AD-E501 470

Copy 15 of 235 copies

AD-A245 433

(2)

IDA PAPER P-2590

THE STATE OF MULTIPLE SENSOR, MULTIPLE TARGET TRACKING IN BALLISTIC MISSILE DEFENSE

Barry E. Fridling

September 1991

Prepared for
Strategic Defense Initiative Organization

IDA
INSTITUTE FOR DEFENSE ANALYSES
1801 N. Beauregard Street, Alexandria, Virginia 22311-1772

IDA Log No. HQ 91-38012

20000831027

DEFINITIONS

IDA publishes the following documents to report the results of its work.

Reports

Reports are the most authoritative and most carefully considered products IDA publishes. They normally embody results of major projects which (a) have a direct bearing on decisions affecting major programs, (b) address issues of significant concern to the Executive Branch, the Congress and/or the public, or (c) address issues that have significant economic implications. IDA Reports are reviewed by outside panels of experts to ensure their high quality and relevance to the problems studied, and they are released by the President of IDA

Group Reports

Group Reports record the findings and results of IDA established working groups and panels composed of senior individuals addressing major issues which otherwise would be the subject of an IDA Report. IDA Group Reports are reviewed by the senior individuals responsible for the project and others as selected by IDA to ensure their high quality and relevance to the problems studied, and are released by the President of IDA.

Papers

Papers, also authoritative and carefully considered products of IDA, address studies that are narrower in scope than those covered in Reports. IDA Papers are reviewed to ensure that they meet the high standards expected of refereed papers in professional journals or formal Agency reports.

Documents

IDA Documents are used for the convenience of the sponsors or the analysts (a) to record substantive work done in quick reaction studies, (b) to record the proceedings of conferences and meetings, (c) to make available preliminary and tentative results of analyses, (d) to record data developed in the course of an investigation, or (e) to forward information that is essentially unanalyzed and unevaluated. The review of IDA Documents is suited to their content and intended use.

The work reported in this document was conducted under contract MDA 903 89 C 0003 for the Department of Defense. The publication of this IDA document does not indicate endorsement by the Department of Defense, nor should the contents be construed as reflecting the official position of that Agency.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
September 1991

3. REPORT TYPE AND DATES COVERED
Final--November 1990 to September 1991

4. TITLE AND SUBTITLE
The State of Multiple Sensor, Multiple Target Tracking in Ballistic Missile Defense

5. FUNDING NUMBERS
C - MDA 903 89 C 0003
T - T-R2-597.01

6. AUTHOR(S)
Barry E. Fridling

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Institute for Defense Analyses
1801 N. Beauregard St.
Alexandria, VA 22311-1772

8. PERFORMING ORGANIZATION REPORT NUMBER
IDA Paper P-2590

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Strategic Defense Initiative Organization
The Pentagon, Room 1E149
Washington, DC 20301-7100

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

This report addresses the latest developments and some of the critical issues pertaining to tracking algorithm development for ballistic missile defense (BMD). For the last several years, tracking has been recognized as one of the most complex and challenging tasks in BMD. During this time tracking algorithm development has been pursued vigorously, with substantial results and continuing progress. Part I of this report has the character of a survey of the latest developments and describes the activities in different organizations and the results obtained in critical technical areas. It describes developments in two areas pertaining to algorithm development and two related to algorithm simulation and evaluation: Data Association; Cluster Tracking; Tracking Simulation; and The Surveillance Testbed (STB). Part II is essentially a technical analysis of issues, problems, and candidate approaches in selected key areas. Some remaining problems pertaining to the design, utilization, and evaluation of tracking algorithms are: Birth-to-Death Tracking; Booster Tracking and Template Matching; Midcourse Track Initiation; and Scoring Methods for Track Performance.

14. SUBJECT TERMS
ballistic missile defense, tracking algorithm, surveillance testbed, multiple target tracking, multiple sensor tracking, birth-to-death tracking

15. NUMBER OF PAGES
105

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	SAR

IDA PAPER P-2590

THE STATE OF MULTIPLE SENSOR, MULTIPLE TARGET TRACKING IN BALLISTIC MISSILE DEFENSE

Barry E. Fridling

September 1991

INSTITUTE FOR DEFENSE ANALYSES

Contract MDA 903 89 C 0003
Task T-R2-597.01

PREFACE

This paper constitutes a deliverable to Task T-R2-597.01, "SDI Battle Management/ C^3 Studies," in accordance with Section 5.0--Schedule--of the task order dated 1 October 1990. The Institute for Defense Analyses (IDA) was tasked by the Strategic Defense Initiative Organization (SDIO) to monitor, evaluate, and facilitate the development of tracking algorithms. This paper undertakes to survey the state of the practice and the state of the art in multiple sensor, multiple target tracking algorithms under development for or applicable to ballistic missile defense in order to ascertain the status of activities in this critical area.

The author would like to gratefully acknowledge the comments of Parney Albright and Albert Perrella, and especially Oliver Drummond, Keh-Ping Dunn, and Gabriel Frenkel.

CONTENTS

TABLES

FIGURES

ABBREVIATIONS

AA	Attack Assessment
BMD	Ballistic Missile Defense
BSTS	Boost Surveillance and Tracking System
C^3	Command, Control, and Communications
CSO	Closely Spaced Object
FFRDC	Federally Funded Research and Development Center
GBI	Ground-Based Interceptor
GPALS	Global Protection Against Limited Strikes
GSTS	Ground-Based Surveillance and Tracking System
IDA	Institute for Defense Analyses
JPDA	Joint Probability Data Association
M^3R	Multi-Dimensional Maximal Marginal Return
MHT	Multiple Hypothesis Tracking
MMR	Maximal Marginal Return
MTT	Multiple Target Tracking
NRL	Naval Research Laboratory
OSM	Object Sighting Message
PBV	Post-Boost Vehicle
RSS	Root-Sum-Square
RV	Reentry Vehicle
SDIO	Strategic Defense Initiative Organization
SEIC	Systems Engineer and Integration Contractor
SSGM	Strategic Scene Generation Model
SSTS	Space-Based Surveillance and Tracking System
STB	Surveillance Testbed
TMD	Theater Ballistic Missile Defense
TW	Tactical Warning

EXECUTIVE SUMMARY

Until recently, most of the Strategic Defense Initiative Organization's (SDIO) tracking research and development focus has been on passive electrooptical sensors housed on orbiting satellites or ground-based rockets launched into sub-orbital trajectories. These sensors, as part of a so-called Phase 1 defense system, were to detect, measure, track, and discriminate large numbers of missile boosters and reentry vehicles (RVs) in a full-force Soviet strategic attack as they fly from boost phase through midcourse to early reentry, in often noisy background clutter environments, closely interspersed during portions of their flights with many escorting decoys. Ground-based radars, which have always been a part of SDIO terminal ballistic missile defense systems, have not been critical items in Phase 1 tracking research and development. The emphasis instead has been on mastering the great challenges of conducting surveillance with sensors that generate angles-only measurements (also known as lines of sight or directions to the target) and intensity measurements during boost and midcourse, to execute intercepts as early after launch as possible. Much of this report is couched in terms of electrooptical sensors.

Recently, SDIO's mission has been expanded to emphasize theater ballistic missile defense (TMD) and defense against accidental or unauthorized strategic missile attacks in a mission known as the Global Protection Against Limited Strikes (GPALS). Radars are critical sensors in TMD and GPALS. SDIO tracking research and development efforts now need to be focused on two issues. One is on executing the ground-based TMD mission and on extending the battle space of TMD systems by utilizing tracking information from satellite-based systems. The other is the full strategic threat and a Phase 1 defense.

This report addresses the latest developments and some of the critical issues pertaining to tracking algorithm development for ballistic missile defense (BMD). For the last several years, tracking has been recognized as one of the most complex and challenging tasks in BMD. During this time tracking algorithm development has been vigorously pursued, with substantial results and continuing progress. Part I of this report is a survey of the latest developments in this area. In spite of this progress, however, many critical issues remain unresolved. Part II of this report is a discussion of those critical issues.

The two parts are complementary. Part I has the character of a survey and describes the activities in different organizations and the results obtained in critical technical areas. Part II is essentially a technical analysis of issues, problems, and candidate approaches in selected key areas. The following major subjects are discussed in the two parts:

Part I: Survey of progress in some key tracking technologies--describes developments in two areas pertaining to algorithm development and two related to algorithm simulation and evaluation:

- Data Association
- Cluster Tracking
- Tracking Simulation
- The Surveillance Testbed (STB).

Part II: Some remaining problems pertaining to the design, utilization, and evaluation of tracking algorithms covers four topics:

- Birth-to-Death Tracking
- Booster Tracking and Template Matching
- Midcourse Track Initiation
- Scoring Methods for Track Performance.

PART I SURVEY OF PROGRESS IN SOME KEY TRACKING TECHNOLOGIES

In Part I, we undertake to provide a sense of where things stand regarding:

- How many targets can be tracked, how well, and in what densities and scenarios?
- In computer simulations to date, how many targets have been tracked, how well, and in what densities and scenarios?
- By what criteria should computer simulations be judged?
- Where is additional work required and what are the critical issues in algorithm development, simulation, and evaluation?

Data Association

Data association, often referred to as scan-to-scan correlation, is the decision process of linking observations and tracks from the same target. Data association is very challenging in a dense observation environment arising from clutter, false alarms, and multiple targets and observations arising from unresolved closely spaced objects (CSOs). An electrooptical sensor's ability to resolve closely spaced objects depends on the properties of the optics and focal plane, the sensor signal processing, and the range and viewing geometry to the targets. An observation from a group of unresolvable closely spaced objects may be indistinguishable from the observation for a single object. A CSO may also appear as a relatively large (compared to the signal from individual objects) clump on the sensor's detectors in what is referred to as an extended object.

Table S-1 shows the algorithms and key features for data association methods being applied to BMD. High observation density is the single most defining characteristic of SDI tracking problems, particularly for Phase 1 defense systems. It is the determinative factor in the selection, implementation, and complexity of data association algorithms.

Table S-1. Data Association Algorithms

Type	Feature	Explanation
Assignment	Coordination	Assignment can be performed on each track independently of all other tracks (locally) or on all the tracks simultaneously in a coordinated fashion (globally).
	Dimension	Assignment can be performed on two data lists, such as one set of tracks and the new frame of observations, or more than two data lists such as one set of tracks and multiple frames of data.
	Number	Assignment is usually unique, such as one observation to one track, but can also be multiple, such as multiple observations to one track or vice versa.
Probabilistic Data Association	Dimension	Tracks are updated by an average over all feasibly associated observations from one or more frames of data.
Multiple Hypotheses	Splitting	Create additional tracks for each feasibly associated observation.
	Observation-oriented	Consider each observation in turn as originating from a new target or a feasibly associated existing track.

For the high observation densities of Phase 1 scenarios, whether good tracking performance can be accomplished at affordable, or for that matter achievable, computational expense very much remains an open question. For TMD and GPALS data association will be less challenging, but, depending on the scenario, will not necessarily be no challenge. Judgments on tracking performance for TMD and GPALS, as for Phase 1, must await high detail, high fidelity, credible simulations.

Cluster Tracking

One innovative approach to managing the high density threat, particularly during early midcourse and Phase 1, is to forgo tracking individual objects and instead to track closely spaced observations as a group or cluster in terms of the mean and extent. Clever modeling was required to devise a filter to track the cluster extent. Cluster tracking raises many issues, not the least of which is that since individual target tracks are ultimately what is required, when is cluster tracking performed rather than individual target tracking and vice versa? Also, when and how is the transition between cluster and individual object tracking accomplished?

The Panels have defined a spectrum of group tracking approaches:

- *Group:* Group properties alone are tracked.
- *Group with Simple Individual:* Simple individual object information is tracked but the tracking of group properties is emphasized.
- *Individual and Simple Group:* Simple group information is tracked but individual object tracking is emphasized.
- *Individual object tracking:* Individual object tracking alone is performed.

The group tracking efforts with which we have some detailed information are described in Table S-2. Much more work needs to be done to explore the diversity of algorithms and algorithm architectures and the critical issues associated with group tracking. Work to date has only begun to address the problems and possibilities.

The Surveillance Testbed (STB) will provide an important environment to investigate these issues. There is one group tracking algorithm in the initial set of test articles being hosted on the STB. The status of group tracking remains for the most part as it was last year: In need of experiments and analyses that explore the diversity of group tracking algorithms and algorithm architectures and the critical issues outlined above.

Table S-2. Cluster Tracking Algorithms

Contractor	Principal Developers	Brief Description
Hughes Aircraft and General Dynamics	Oliver Drummond, Samuel Blackman	Multiple Sensor algorithm for tracking cluster centroid and extent.
MIT/Lincoln Laboratory	C.B. Chang, Keh-Ping Dunn, Ming Tsai	Individual tracks for the observations defining the edge of a cluster to form tracks for cluster members, which are assumed to be moving in parallel with edges.
Nichols Research Corporation	Robert Osterheld, Lisa Brahm	Develop individual object tracks of RVs and decoys deployed off a common PBV through their closely spaced phase.
Teledyne Brown	Keith Maples	Track cluster centroid and the parameters of an RV-decoy deployment model to develop individual object tracks.

Tracking Simulations

Observation density is also the single most defining characteristic in SDI tracking simulations, particularly Phase 1 simulations. Tracking simulations measure computational performance (speed and memory) and tracking performance (estimation and association performance measures such as estimation accuracy, credibility, and reliability and track purity). As a practical matter, many simulations emphasize one aspect over the other. Simulations with larger and more dense threats are more often used to investigate computational performance because the data association algorithm and tracking filter design, implementation, and experimentation are compromised by insufficient computational resources. Those data association algorithms and tracking filters are selected more for their computational thriftiness than for their tracking performance. High observation density indicates the level of ambitiousness and complexity in the simulation, determines the adequacy of computer resources, and must be considered in the measurement of the level of difficulty and in the judgement of the quality of tracking performance.

Good computational performance by itself is necessary but not sufficient. The same is true for estimation and association performance. Ten thousand objects have been tracked in real-time but without much attention to estimation and association performance.

In one simulation 460 midcourse objects in moderate Phase 1 density were tracked and in another simulation 125 midcourse objects in heavy Phase 1 density were tracked--both with good estimation performance, given crucial, unverified, and very optimistic assumptions concerning the quality of boost-phase tracks handed over to the midcourse tracking algorithms.

When one considers tracking simulations four issues should be kept in mind:

- *How challenging is the threat being tracked?* Tracking simulations can be likened to diving and ice skating competitions, for instance, in the sense that performance must be evaluated in part by the level of difficulty of the effort.
- *What are the important assumptions and initial conditions?* This is something of an extension to assessing the level of challenge of the threat. There are several key concerns: To what extent are clutter and background effects included and how are they modeled? What is assumed for sensor properties such as resolution and accuracy? How is the sensor and signal processing modeled? Is sensor tasking scripted in advance or performed on-line? Are data misassociations permitted? Start-up conditions can grossly affect tracking performance. For midcourse simulations the key issue concerns how midcourse tracks are initiated: Does the simulation rely on tracks handed over from boost phase and if so how good are those track hand-overs?
- *What are the key algorithm details?* It is important to know some detail of the algorithms and algorithm architecture to appreciate the context of the work.
- *What are the scoring methods and measures of effectiveness?* How are tracks associated with true targets for evaluation?

Table S-3 lists some published results of tracking simulations. For reasons that include competitiveness, not all simulation results are published. We list only those simulations where we have some moderate sense of the answers to the four questions above.

Surveillance Testbed

Substantial progress has been made towards construction of a testbed to experiment on and evaluate alternative surveillance algorithms--detection, tracking, discrimination, and sensor tasking--and to assess ballistic missile defense system-level surveillance performance. While the STB is being constructed to support both purposes, to the extent that one takes precedence over the other, first priority must go to evaluation of surveillance algorithms: an emphasis in favor of experimentation on algorithms increases the prospects for a testbed that successfully assists in the development of surveillance algorithms and accurately assesses system-level surveillance performance. The program seems on track to ensuring the creation of a facility that is sorely needed by the SDI tracking community.

Table S-3. Midcourse and Birth-to-Death Tracking Simulations

Organization	Number and Density of Threat Objects	Simulation Objective
Advanced Systems Architectures	460; moderate density	Investigate tracking performance
Alphatech	6,260	Investigate computational performance
	125 (1 booster)	Investigate tracking performance
	50	Investigate performance of various assignment algorithms
ESL	460; moderate density	Investigate tracking and computational performance
GSTS	648	Investigate tracking performance
NTBIC	Subsets of DTT-1 of various densities	Investigate computational and tracking performance
Sandia National Laboratories	10,000	Investigate computational performance

The Surveillance Testbed is critical to SDIO. The STB will be the SDIO facility where contractors can with a minimum of modification run their own software in high fidelity, high detail surveillance simulations to verify detection, tracking, discrimination, and sensor tasking. A contractor's algorithms can be tested by themselves on the STB as well as in their role in a complete surveillance system by inserting them into a complementary set of "house algorithms," algorithms maintained on the STB. With SDIO approved threat scenarios, and SDIO supplied signal generation and sensor and signal processing data that are the inputs to the surveillance algorithms, critical algorithm experimentation and evaluation can take place in a controlled environment and scored according to standardized methods, thereby facilitating performance comparisons. By having the STB supply the inputs that drive high fidelity, high detail simulations, SDIO saves the resources otherwise spent when each contractor develops their own extensive simulation drivers.

The SDI Tracking Panels have played a critical role in the development of the STB. Before the STB program was started the members of the Tracking Panels, representing tracking algorithm designers from across government, FFRDCs, contractors and academe, called for an algorithm testbed of this sort. A testbed on which experimentation could be performed and high detail, high fidelity threat scenarios complete with sensor measurement data could be produced and supplied to contractors for use in their own testbeds. During the development of the STB, the Panels provided a peer review of the STB development plans and supplied the input of the intended user community. The Panels deserve credit for helping to shape the STB into a facility that should well serve SDIO and its contractor community.

The Panels identified and analyzed three different critical interface issues for the STB:

- Interfaces between individual test articles and the STB test environment;
- Interface requirements stemming from permitting feedback from tracking algorithm to sensor and signal processor, such as sensor tasking; and
- Interfaces between test articles.

It should be no surprise that hosting tracking algorithms on the STB will require some software modifications. Everyone can agree that it is imperative to keep the modifications to a minimum. The real issues are what sort of modifications, how many, and by whom, the STB contractor or the tracking algorithm contractor? For the most part, it must be the tracking algorithm developer who modifies their software since they are the most knowledgeable of the code's contents. But their willingness to utilize the STB depends on the scope of the modifications. The STB's government sponsors need to appreciate that, everything else in order, the STB will fail or succeed based on the scope of software changes required for hosting surveillance algorithms.

The Panels' recommendations to the STB are summarized as:

- Development of flexible interfaces between STB test environment and tracking algorithms must be emphasized early in order to best ensure that the STB achieves its goal of providing a testbed for developing and evaluating alternative surveillance algorithms;
- Limited emphasis should be given to early results of system-level performance experiments with the "representative, baseline" algorithms used for testbed integration validation;
- The STB must support a portable testbed facility.

These testbed priorities should be clearly established:

- Vigorous surveillance algorithm development must precede surveillance system evaluation;
- Prepare early to accommodate the diversity of tracking algorithms and algorithm architectures;
- Interfaces between test environment and tracking algorithms must be flexible and robust;
- Only after representative surveillance performance is appreciated and quantified by experiments on individual surveillance algorithms can system-level surveillance performance be assessed accurately.

PART II. SOME REMAINING PROBLEMS PERTAINING TO THE DESIGN, UTILIZATION, AND EVALUATION OF TRACKING ALGORITHMS

Birth-to-Death Tracking

Ballistic missile defense birth-to-death tracking is the concept of maintaining continuous tracks on targets from launch through to impact by fusing tracks across sensor elements. Birth-to-death tracking schemes range from the grand to the temperate. In the grandest design of birth-to-death tracking, downstream narrow field-of-view sensors that lack adequate independent search capability are enabled by pointing directions from upstream sensors. Additionally, downstream sensors initialize tracks by relying entirely on upstream track hand-overs. Boost-phase tracks would be handed over to the post-boost and midcourse surveillance sensors for continuation: booster burnout position and velocity would initialize the post-boost vehicle (PBV) track. Narrow field-of-view optical midcourse sensors would be directed where to look for the PBV rather than independently searching. Ground-based radars would also be cued where to look for midcourse and re-entering objects to gain increased detection ranges. As the PBV dispenses reentry vehicles (RVs) and decoys, tracks for each would be established by spawning new tracks from the continuing PBV track. Every object in the midcourse then could be traced back to its origin, PBV and booster, and a track for each established essentially by continuation of booster tracks. There would never be need during midcourse for the "cold start" track initiation procedures of assembly of a sequence of measurements for initial orbit determination data processing. All midcourse tracks would be initialized by "warm start" track initiation: hand-over of track data from upstream sensor elements and spawning new tracks from existing tracks.

Track hand-overs from boost and midcourse to ground-based defense systems could enable a number of enhancements in their performance and battle space, including

- Cues to the ground-based radars to concentrate their search into narrow fields-of-view to gain increased detection ranges;
- Early commit of ground-based interceptors before their radars see the targets but the radars would guide the interceptors as the targets come into view; and
- Launch of ground-based interceptors entirely independent of their radars where tracks from space-based sensors provide in-flight target updates that enable on-board guidance algorithms to fly the interceptors into the close proximity of their targets, where the interceptor's own on-board sensors would take over.

Birth-to-death tracking in its grandest design is logical and efficient. Such schemes are ambitious, perhaps even feasible. But, as Table S-4 indicates, there are significant liabilities in these approaches.

Table S-4. Key Features of Birth-to-Death Tracking

Advantages	Liabilities
Omniscient accounting of threat objects	Timely, seamless handover required
Enabling of narrow field-of-view sensors	Susceptibility to catastrophic failure
Facilitates track initiation	
Reduce computational requirements	

Temperate birth-to-death tracking schemes with less than absolute reliance on seamless link between sensors and sensor elements are directed towards:

- Obviating cold start track initiation if upstream tracks are available;
- Assisting rather than enabling relatively narrow field-of-view sensors;
- Assisting rather than supplanting individual sensors and sensor elements.

Ultimately, it is a question of capability versus robustness against catastrophic failure. Each sensor element and each individual sensor must be capable of searching a reasonable surveillance region and performing cold start track initiation to reduce vulnerability and to constitute a system that works in a world of communications delays and misdirections and the unexpected, but only if adequate capability is achievable or affordable in individual sensors or sensor elements.

Booster Tracking and Template Matching

Boost-phase tracking experience to date is limited. What can be performed well is the tracking of a small number of targets across the focal plane (focal plane tracking) of a passive electrooptical sensor on-board geostationary satellites without explicitly modeling the boosters' detailed three-dimensional dynamics or explicitly tracking the boosters' three-dimensional trajectory. Historical averages and *a priori* assumptions for booster altitude versus ground range flight profiles and intensity versus time profiles, known as templates, are used to estimate launch locations, times, and azimuths, and to identify missile types. But with regard to using focal plane tracks and templates to predict future states in post-boost, midcourse, and reentry, the accuracy required for BMD has not yet been demonstrated. Brilliant Pebble booster tracking of a particular sort has been demonstrated in limited simulations with low fidelity data and without clutter. Extensive simulations that demonstrate convincingly the tracking accuracy required for booster surveillance, good weapon-to-target assignment, and good pebble guidance performance remain to be done. In this report we review critical issues in, and methods of, booster tracking and template matching.

Booster tracking algorithms can:

- Estimate so-called tactical parameters, which are the missile launch locations, times, azimuths, altitudes, and the degree to which the missile is lofted or depressed from a nominal trajectory;
- Assess the number and types of missiles launched in the raid;
- Predict missile payload impact points on the earth;
- Cue midcourse and terminal ballistic missile defense systems, both sensors and interceptors; and
- Provide fire control information for booster interception.

Tactical parameter estimation, raid assessment, and coarse impact point prediction constitute the traditional tactical warning and attack assessment (TW/AA) functions. More precise impact point predictions and cues to midcourse and terminal sensors and interceptors can enhance midcourse and terminal BMD performance and also enhance TW/AA performance.

As shown in Table S-5, there are essentially three data processing or filtering methods for tracking boosters.

Table S-5. Booster Tracking Filter Methods

Method	Critical Issues
Kalman Filter	Modeling and integration of three-dimensional booster dynamics should give best performance if viable, reliable, and credible. Need to compensate for model errors, including maneuvers.
Line-of-sight triangulation	Data association of lines-of-sight in a dense observation environment. Accuracy of velocity estimates is limited by the interpolation and numerical differentiation procedures.
Least squares (Template matching)	Use of historical averages and *a priori* assumptions for altitude versus ground range flight profiles (templates) which also assumes trajectories for altitude are two dimensional. Convergence conditions and accuracy and the effects of clutter, false alarms, and multiple targets.

Two general fundamental issues attend the use of historical averages and *a priori* assumptions on booster altitude versus ground range profiles and assumed two-dimensional flight trajectories:

- To which template should the sensor data be matched?
- What are the reliability and credibility of relying on these assumptions?

There are essentially four sources of templates:

- The Master Target Model Book, published by the Aerospace Corporation, contains templates for altitude, ground range, intensity, mass, thrust, etc. These profiles, typically one per missile type and mod, are from reconstructed trajectories using data observed by national technical means on systems that have already flown.
- The SDIO Threat generation community, apparently so far in an ad hoc manner, has produced to specification a family of templates representing lofting and depression for the future missile systems that typically populate SDIO threat scenarios. It is important to emphasize these are not based on reconstructed trajectories.
- Contractors with sophisticated booster modeling capabilities augmenting whatever templates they are given from whatever source to create lofted and depressed templates.

- Averages over threat trajectories in a particular attack scenario on a particular threat tape.

Clearly, use of the last sort of template is suspect. The immediate relationship of the booster trajectories in the threat with the templates that are then used to track them, the very same trajectories that generated the templates, produces possibly invalid results and grossly misleading performance assessments. This source of templates must be strongly discouraged.

The Master Target Model Book cannot supply the templates for SDIO scenarios set in the future because of the mismatch between templates for missile systems that have been observed, on the one hand, and SDIO threat scenarios that involve future missile systems that have yet to fly and for which no templates based on observed data can be constructed, on the other. The community that produces the Model Book could be asked to produce templates for future systems but this would represent a departure from their standard methods, most importantly the reliance on observed data. SDIO, possibly by way of its Threat Working Group, possibly in conjunction with the intelligence community, needs to firmly control the development and promulgation of templates for use by SDIO contractors. There currently is a gap of immense proportions between the intelligence community providing templates on current inventory missiles and the free-for-all of assumptions on the character and content of template data being made by SDIO contractors. If control is not taken by SDIO, the validity of all template matching results is at risk and could be considered suspect.

The most important issue is the degree to which templates are identical to the trajectories being tracked. Should they match? If the altitude versus ground range templates closely parallel the booster trajectories to be tracked then good template matching performance should be expected. In a sense, close identity is akin to having assumed away the problem: All the uncertainty in the booster motion is removed and captured in the *a priori* data bases of templates. Unless we believe we can assume away the problem, templates, in general, should not be identical to the actual booster trajectories to be tracked. One exception might be third world missile forces whose unsophisticated guidance systems would keep the booster trajectories to simple flight profiles.

The bottom line is that reliance on *a priori* information may leave the algorithm vulnerable to boosters that do not do the expected, or the mean, or are of type and circumstance outside the *a priori* information's domain of applicability. The consequence may be susceptibility to catastrophic failure. Balance is the key: use *a priori* information when necessary but in a manner that is as flexible as possible and that does not leave undue susceptibility to catastrophic failure.

Midcourse Track Initiation

Midcourse angles-only initial orbit determination is an old subject with a pedigree that extends back almost two hundred years to Gauss and Laplace. The fundamental midcourse tracking challenge to this time has been the large number and high density of missile booster, RVs, decoys, and clutter to be tracked. As the severity of the threat declines, as measured by density of objects seen by a sensor, from a defense against a full Soviet attack, to a defense against accidental or unauthorized missile launches, to theater missile defense, data association becomes less of a concern. Single satellite angles-only track initiation in the dense observation environments of Phase 1, and perhaps GPALS, is a critical issue.

The issues in BMD cold start track initiation, both for the template matching in boost-phase and for midcourse, are first the ability, in a dense observation environment, to assemble a reasonably small number of credible time sequences of angles-only measurements without being able to use models for the detailed models for the three-dimensional target motion. The other issue is the reliability of the initial trajectory/orbit determination algorithms and the accuracy, precision, and credibility of their initial state estimates and estimation errors. Low precision state estimation errors will lead to greatly complicated data association problems for track maintenance in dense observation environments. The critical issue is to gain high enough precision estimation errors to mitigate data association problems. We know of no simulations where these issues are thoroughly examined.

Scoring Methods for Tracking Performance

Ultimately, the performance of tracking algorithms is judged by the success, or failure, of the mission they support. But it is also important to evaluate tracking algorithms in computer simulations to diagnose and evaluate their performance. Evaluation of tracking performance is straightforward in an environment of few, widely spaced targets and no false alarms or clutter. In this sparse environment, a track is consistently updated with measurements from the same target. The track, or state estimate, is then associated and compared with the true state of the target, which is obvious as identified by the one source of the measurements.

Performance evaluation is more complex in a dense environment of:

- False alarms;
- Clutter;
- Multiple targets;
- Individual observations arising from unresolved closely spaced objects (CSOs).

With misassociations and unresolved CSOs, the source of the measurements in a track will not be a clear indication of a single target, thus confusing which track is to be compared with the true state of a target. Furthermore, in a dense environment, there may be

- Missed tracks: targets without tracks;
- Redundant tracks: more than one track for one target;
- Spurious tracks: tracks for no targets whatsoever.

Scoring tracking algorithm estimation and association performance has been a major issue in SDI simulations. Some algorithms may generate many "extra" tracks, such as in multiple hypothesis tracking, but the track purity and state estimation accuracy of the N best are better than the N tracks of algorithms that do not knowingly generate "extra" tracks, such as local nearest neighbor. Insofar as track purity and state estimation are concerned, the former is to be preferred, whereas the latter may be preferred from the standpoint of computational and memory costs and size, weight, and power of on-board processors.

These methods were initially developed by individuals and further developed and adapted by the members of the SDI Panels on Tracking. It is part of an ongoing process and is not to be considered as the last word on the subject.

Track purity over a time interval refers to the degree to which a track's measurements over that time originate from a particular target. In single target tracking without false alarms and clutter, track purity is ensured and the association of track-to-truth unambiguous. Multiple target tracking typically involves many impure tracks and, therefore, ambiguous track-to-truth association. We will define scoring criteria for track purity in dense target environments. In principle, track purity can be used to determine track-to-truth associations but in dense target environments and for some MTT algorithms the concept of track purity loses some of its meaning.

We suggest a method for track-to-truth association based on a global nearest neighbor assignment approach. At each of the designated evaluation times, a global nearest neighbor assignment algorithm is executed to *uniquely* associate tracks and targets. After tracks and truth have been asscciated, we can evaluate performance criteria for the two functions of a multiple target tracking algorithm:

1. *Data association.* This function selects the observations to be used by the track filter to update the state estimate. Its measures of effectiveness will be track purity and misassociation. They measure the consistency with which a track is updated with measurements from a single target or a set of targets, respectively; and
2. *Estimation.* This function transforms sensor measurements into estimates of the target's state, usually the target's trajectory described by position, velocity, acceleration, etc, and the target's state estimation error covariance. The distance between the state estimate and the true state and the credibility of the filter calculated covariance matrix measure the performance of the tracking filter, which is affected by data misassociation and other errors.

The goal of the detailed scoring methods for tracking algorithm estimation and association performance is to provide a guide or standard with which all tracking algorithms can be evaluated. This report presents formulas and criteria for many of the major functions in tracking adapted by the Panels.

1. INTRODUCTION

This report undertakes to characterize the state of the practice and the state of the art of multiple sensor, multiple target tracking algorithms in ballistic missile defense (BMD). Until recently, most of the Strategic Defense Initiative Organization's (SDIO) tracking research and development focus has been on passive electrooptical sensors housed on orbiting satellites or ground-based rockets launched into sub-orbital trajectories. These sensors, as part of a so-called Phase 1 defense system, were to detect, measure, track, and discriminate large numbers of missile boosters and reentry vehicles (RVs) in a full force Soviet strategic attack as they fly from boost phase through midcourse to early reentry, in often noisy background clutter environments, closely interspersed during portions of their flights with many escorting decoys. For simplicity, we will refer to both of these basing modes as satellite-based, in keeping with the sensor's location while tracking.

Ground-based radars, which have always been a part of SDIO terminal ballistic missile defense systems, have not been critical items in Phase 1 tracking research and development. The emphasis instead has been on mastering the great challenges of conducting surveillance with sensors that generate angles-only measurements (also known as lines-of-sight or directions to the target) and intensity measurements during boost and midcourse, to execute intercepts as early after launch as possible. Recently, SDIO's mission has been expanded to emphasize theater ballistic missile defense (TMD) and defense against accidental or unauthorized strategic missile attacks in a mission known as the Global Protection Against Limited Strikes (GPALS). Radars are critical sensors in TMD and GPALS. SDIO tracking research and development efforts now need to be focused on two issues. One is on executing the ground-based TMD mission and on extending the battle space of TMD systems by utilizing tracking information from satellite-based systems. The other is the full strategic threat and a Phase 1 defense.

1.1 INTRODUCTION TO TRACKING

A track is an estimate, based on sensor measurements, of the position and velocity and sometimes acceleration and key parameters that describe the target's trajectory and properties, such as mass, temperature, etc. The position, velocity, acceleration, and any key parameters are collectively referred to as the target's state. Unless otherwise specified, we will usually understand target state to refer to a target's position, velocity, and acceleration, that is, trajectory. A track, then, is an estimate of the target's current and future state derived from sensor measurements of it.

A tracking algorithm is a sequence of logical and mathematical procedures for: (1) associating sensor measurement data from the same target across multiple frames of data and across multiple satellites or associating tracks of the same target across multiple satellites; and (2) processing that data into estimates of current and future target position, velocity, and acceleration. Association and estimation are interrelated procedures usually based on detailed physics models for the forces governing the target's trajectory and detailed models for the relationship between sensor measurements and the target's state. Sensor measurement data are the inputs to a tracking algorithm; estimates of the target trajectory are the outputs. The estimation algorithm establishes criteria by which the association algorithm chooses subsequent measurements for subsequent data processing by the estimation algorithm.

The two most important data processing algorithms that transform sensor measurement data into trajectory estimates are Kalman filters and least square filters. A Kalman filter is an algorithm that, based on models for target dynamics and sensor measurements, generates a state estimate and state estimation error covariance at the current time using all the sensor measurement data to that time. The estimate and its error are updated each time subsequent measurement data become available, in what is referred to as recursive or sequential processing. The filter also predicts future values for the state estimate and state estimation error covariance without further data. Kalman filters are an optimum method for processing the data according to the statistical measure of optimality known as minimum mean square error. In contrast, the least squares filter, in what is referred to as batch processing, simultaneously processes an assembly (time sequence) of data to produce a state estimate and estimation error covariance at a time during the observation interval. The least squares filter is optimum according to the criterion of minimizing the squared errors between the sensor measurement data and a model for the target motion that produced that data.

A passive electrooptical observation measures azimuth and elevation and cannot measure range to a target. Azimuth and elevation define a line-of-sight vector that describes the direction to a target from the sensor. A target's position with respect to the sensor is specified by the product of range and line-of-sight vector.

In passive electrooptical sensors, tracks are either focal plane tracks or full state tracks. A focal plane track typically uses a Kalman filter to estimate a target's trajectory across the focal plane without explicitly modeling its dynamics in three dimensions. The sensor's azimuth and elevation sensor measurement data are used to estimate the current and future azimuth and elevation position, velocity, and acceleration. Range and its derivatives are neither measured nor estimated.

In single satellite passive electrooptical tracking algorithm approaches, focal plane tracking would be followed by an initial trajectory/orbit determination algorithm. This is often, but not always, a least squares filter to batch process the entire time sequence of azimuth-elevation measurements in a focal plane track. In general, initial trajectory/orbit determination algorithms transform the essentially two-dimensional angles-only information of a focal plane track (azimuth and elevation) into initial three-dimensional information on the full state, which can initialize a full state Kalman filter. Based on models describing a target's three-dimensional dynamics and the sensor measurement process, the full state Kalman filter maintains estimates for three-dimensional position, velocity, and acceleration, alternatively updating estimates at the time each new sensor measurement data become available with predicting estimates ahead to future times.

If another satellite is available and observes the same surveillance region, its azimuth-elevation measurement data can be associated and combined with the first satellite's angles-only measurement data to produce range measurements to the targets by triangulation. One satellite's focal plane tracks can similarly be associated and combined with those of another satellite to produce range data. Stereo data processing would then follow.

Stereo data processing algorithms for fused line-of-sight data come in two general types. One type implements an initial trajectory/orbit determination algorithm, different in detail from the angles-only algorithms in the single satellite case, but with the same goal: to determine the three-dimensional trajectory information not directly available in the measurement data, namely, three-dimensional velocity and acceleration. Three-dimensional tracks would then be maintained by a Kalman filter, either in place of the individual satellite's focal plane tracks or in addition to them.

The other type of stereo data processing algorithm obviates dynamical modeling requirements by assembling a time sequence of three-dimensional target positions from continually fused focal plane tracks maintained by the stereo partner satellites. Three-dimensional target velocity at each time is computed by interpolation and numerical differentiation of the positions.

A major activity in all tracking algorithms, single satellite as well as multiple satellite, is the association of a single frame of observation data to tracks. Often the first step in association is to eliminate unlikely observation-track pairings in a process known as gating. A gate is a region in the sensor's field of view determined by the Kalman filter where the subsequent measurement originating with a target being tracked is likely to fall. In order to be associated with a particular track, it is necessary but not sufficient for an observation to fall within the gate for that track. Observations from other targets, false alarms, or clutter may also fall within the gate or the target may not have been detected within the gate. When there is more than one observation in a gate or an observation simultaneously falls in more than one gate, there is uncertain association of observations and tracks.

In general, the association problem is one of the most computationally intensive and critical aspects of tracking, particularly in dense observation environments. Incorrect observation-to-track association can lead to: poor track performance, that is, a large difference between estimated and true target trajectories; loss of track as the filter follows an incorrect sequence of observations; and tracking errors far worse in reality than those predicted by the Kalman filter. High tracking precision mitigates the association problem by generating smaller gates. Errors incurred during multiple satellite observation and focal plane track association also have very serious effects on performance. These associations are greatly complicated by multiple intersections of lines of sight (ghosting) or non-intersecting lines of sight.

Data association is the most critical challenge in Phase 1 in the dense observation environment arising from clutter, false alarms, and multiple targets. With declining threat severity, from Phase 1 to GPALS to TMD, data association becomes less of a concern in midcourse but by no means does it become of no concern. Data association almost certainly remains a critical issue in boost-phase tracking.

Filter design, modeling, and numerical implementation in order to achieve good performance, credibility, and reliability, when challenged by complex target dynamics, by difficult data association, or by limited sensor information are critical issues that span Phase 1, GPALS, and TMD. For single satellite tracking of boosters in powered flight, clever modeling and design are necessary to track these very dynamic targets in a probably dense observation environment arising from clutter and false alarms using the limited sensor information afforded by angles-only observations.

One innovative approach to managing the high density Phase 1 threat, particularly during early midcourse, is to forgo tracking individual objects and instead to track closely spaced observations as a group or cluster in terms of the cluster mean and extent. Clever modeling was required to devise a filter to track the cluster extent. Cluster tracking raises many issues, not the least of which is that since individual target tracks are ultimately what is required, when is cluster tracking performed rather than individual target tracking and vice versa? Also, when and how is the transition between cluster and individual object tracking accomplished?

1.2 INTRODUCTION TO THIS REPORT

This report is divided into two parts, with Part I a survey for program managers and defense decision makers to provide them a sense of where things stand regarding:

- How many targets can be tracked, how well, and in what densities and scenarios?
- In computer simulations to date, how many targets have been tracked, how well, and in what densities and scenarios?
- By what criteria should computer simulations be judged?
- Where is additional work required and what are the critical issues in algorithm development, simulation, and evaluation?

Towards this end, Part I consists of a survey, including the state of several critical issues in tracking algorithm development:

- Data Association
- Cluster Tracking
- Tracking Simulation
- The Surveillance Testbed (STB).

Tracking simulations measure computational performance (speed and memory) and tracking performance (estimation and association performance measures such as estimation accuracy, credibility, and reliability and track purity). As a practical matter, many simulations emphasize one aspect over the other. Simulations with larger and more dense threats are more often used to investigate computational performance because the data association algorithm and tracking filter design, implementation, and experimentation are compromised by insufficient computational resources. Those data association algorithms and tracking filters are selected more for their computational thriftiness then for their tracking performance. High observation density indicates the level of ambitiousness and complexity in the simulation, determines the adequacy of computer resources, and must be considered in the measurement of the level of difficulty and in the judgement of the quality of tracking performance.

The Surveillance Testbed is critical to SDIO. The STB will be the SDIO facility where contractors can with a minimum of modification run their own software in high fidelity, high detail surveillance simulations to verify detection, tracking, discrimination, and sensor tasking. A contractor's algorithms can be tested by themselves on the STB as well as in their role in a complete surveillance system by inserting them into a complementary set of "house algorithms," algorithms maintained on the STB. With SDIO approved threat scenarios, and SDIO supplied signal generation and sensor and signal processing data that are the inputs to the surveillance algorithms, critical algorithm experimentation and evaluation can take place in a controlled environment and scored according to standardized methods, thereby facilitating performance comparisons. By having the STB supply the inputs that drive high-fidelity, high-detail simulations, SDIO saves the resources otherwise spent when each contractor develops their own extensive simulation drivers.

The SDI Tracking Panels have played a critical role in the development of the STB. Before the STB program was started the members of the Tracking Panels, representing tracking algorithm designers from across government, FFRDCs, contractors, and academe, called for an algorithm testbed of this sort. A testbed on which experimentation could be performed and high-detail, high-fidelity threat scenarios complete with sensor measurement data could be produced and supplied to contractors for use in their own testbeds. During the development of the STB, the Panels provided a peer review of the STB development plans and supplied the input of the intended user community. The Panels deserve credit for helping to shape the STB into a facility that should well serve SDIO and its contractor community.

Part II of this report contains discussions of some remaining problems pertaining to the design and utilization of tracking algorithms:

- Birth-to-Death Tracking
- Booster Tracking and Template Matching

- Midcourse Track Initiation
- Scoring Methods for Track Performance.

In a fully integrated ballistic missile defense surveillance system, tracks would be disseminated and fused throughout the various battle managers and sensor elements: boost phase surveillance sensors, space-based and ground-based space surveillance sensors, and terminal phase sensors. Ballistic missile defense birth-to-death tracking is the concept of maintaining continuous tracks on targets from launch through to impact by fusing tracks across sensor elements. Birth-to-death tracking schemes range from the grand to the temperate.

In the grandest design of birth-to-death tracking, downstream narrow field-of-view sensors that lack adequate independent search capability are enabled by pointing directions from upstream sensors. Additionally, downstream sensors initialize tracks by relying entirely on upstream track hand-overs. Boost-phase tracks would be handed over to the post-boost and midcourse surveillance sensors for continuation: booster burnout position and velocity would initialize the post-boost vehicle (PBV) track. Narrow field-of-view optical midcourse sensors would be directed where to look for the PBV rather than independently searching. Ground-based radars would also be cued where to look for midcourse and re-entering objects to gain increased detection ranges. As the PBV dispenses reentry vehicles (RVs) and decoys, tracks for each would be established by spawning new tracks from the continuing PBV track. Every object in the midcourse then could be traced back to its origin, PBV and booster, and a track for each established essentially by continuation of booster tracks. There would never be need during midcourse for the "cold start" track initiation procedures of assembly of a sequence of measurements for initial orbit determination data processing. All midcourse tracks would be initialized by "warm start" track initiation: hand-over of track data from upstream sensor elements and spawning new tracks from existing tracks.

Birth-to-death tracking in its grandest design is logical and efficient. It possesses the virtues of omniscient accounting of threat objects, enabling of relatively inexpensive narrow field-of-view optical sensors, and avoidance of the immense computational expense and complication of cold start track initiation. But it counts on the existence of a seamless link across sensor elements, in which upstream track information is available exactly when and where it is needed. Such a link is ambitious, perhaps even feasible. But a surveillance system that is entirely reliant upon it is critically susceptible to catastrophic failure.

Temperate birth-to-death tracking schemes cue downstream sensors to assist (rather than enable) relatively narrow field-of-view optical sensors, to increase the battle space of ground-based BMD systems, and to avoid cold start track initiation but only if and when upstream track hand-overs are available. Hand-over tracks are not considered to supplant an independent operational capability for each individual sensor or individual sensor element.

Ultimately, it is a question of capability versus robustness against catastrophic failure. Each sensor element and each individual sensor must be capable of searching a reasonable surveillance region and performing cold start track initiation to reduce vulnerability and to constitute a system that works in a world of communications delays and misdirections and the unexpected. But, only if adequate capability is achievable or affordable in individual sensors or sensor elements.

Boost-phase tracking experience to date is limited. What can be performed well is the tracking of a small number of targets across the focal plane (focal plane tracking) of a passive electrooptical sensor on-board geostationary satellites without explicitly modeling the boosters' detailed three-dimensional dynamics or explicitly tracking the boosters' three-dimensional trajectory. Historical averages and *a priori* assumptions for booster altitude versus ground range flight profiles and intensity versus time profiles, known as templates, are used to estimate launch locations, times, and azimuths, and to identify missile types. But with regards to using focal plane tracks and templates to predict future states in post-boost, midcourse, and reentry, the accuracy required for BMD has not yet been demonstrated. Brilliant Pebble booster tracking of a particular sort has been demonstrated in limited simulations with low fidelity data and without clutter. Extensive simulations that demonstrate convincingly the tracking accuracy required for booster surveillance, good weapon-to-target assignment, and good pebble guidance performance remain to be done. In this report we review critical issues in, and methods of, booster tracking and template matching.

Single satellite angles-only track initiation in the dense observation environments of Phase 1, and perhaps GPALS, is a critical issue. The issues in ballistic missile defense are first the ability, in a dense observation environment, to assemble a reasonably small number of credible time sequences of angles-only measurements without being able to use models for the detailed three-dimensional motion of the targets. The other issue is the reliability of the initial trajectory/orbit determination algorithms and the accuracy, precision, and credibility of their initial state estimates and estimation errors. Low precision state estimation errors will lead to greatly complicated data association problems for track maintenance in dense observation environments. The critical issue is to gain high enough precision estimation errors to mitigate data association problems. We know of no simulations where these issues are thoroughly examined. In this report we discuss in detail approaches to track initiation, booster initial trajectory determination, and to midcourse initial orbit determination.

Ultimately, the performance of tracking algorithms is judged by the success, or failure, of the mission they support. The destruction of a target by an interceptor guided, in part, by tracking information provides one vivid, obvious measure of success. But, what if the interceptor missed? Did the tracking algorithm perform poorly, or the guidance algorithm, or the sensor and signal processing, or the rocket motor?

Scoring tracking algorithm estimation and association performance has also been a major issue in SDI simulations. Succinctly, in a dense observation environment, where the track is made up of observations from many targets, it is non-trivial to decide on the association between tracks and with the true target states that produced those observations. In this report, a method is proposed to associate tracks with true target states. This method was first developed by a member of the Tracking Panels and augmented and adapted by the Panels. After making the association of track to truth, there remains the issue of standard scoring computations and criteria. We derive detailed scoring methods for tracking algorithm estimation and association performance. The goal is to provide a guide or standard with which all tracking algorithms can be evaluated. This report presents formulas and criteria for many of the major functions in tracking adapted by the Panels.

PART I

SURVEY OF PROGRESS IN SOME KEY TRACKING TECHNOLOGIES

2. SURVEY OF DATA ASSOCIATION ALGORITHMS

Data association, also referred to as data correlation, is the decision process of linking observations and tracks from the same target. We will use the term data association for what is often referred to as scan-to-scan correlation. In general, the association could be sensor observations to sensor observations, either near simultaneous observations from multiple sensors or a time sequence of observations from a single sensor. The association could be tracks to tracks from multiple sensors. And the association could also be observations to tracks for one sensor operating independently. Scan-to-scan may imply that the problem is restricted to linking observations with observations.

While association and correlation are synonymous the latter also refers to a specific mathematical quantity or operation. It is useful to point out that the mathematical correlation operator plays no role in establishing links in multiple sensor, multiple target tracking. Finally, to be consistent with strict usage, we will use the term frame instead of scan where a frame is defined as one data collection survey of the surveillance region. With this definition frame is independent of whether the sensor surveils by mechanically sweeping the field of view with detectors or surveils electronically with staring detectors.

Data association is very challenging in a dense observation environment arising from clutter, false alarms, and multiple targets and observations arising from unresolved closely spaced objects (CSOs). The sensor's ability to resolve closely spaced objects depends on the properties of the optics and focal plane, the sensor signal processing, and the range and viewing geometry to the targets. An observation from a group of unresolvable closely spaced objects (CSOs) may appear identical to the observation for a single object. A CSO may also appear as a relatively large (compared to the signal from individual objects) clump on the sensor's detectors in what is referred to as an extended object.

High observation density is the single most defining characteristic of SDI tracking problems, particularly for Phase 1 defense systems. It is the determinitive factor in the selection, implementation, and complexity of data association algorithms. For Phase 1 defense whether good tracking performance can be accomplished at affordable, or for that matter achievable, computational expense remains an open question. For TMD and GPALS data association will be less challenging, but, depending on the scenario, will not necessarily be no challenge.

This chapter, an abbreviated, updated version of similar material in last year's survey, provides a review of the three classes of data association algorithms: assignment, probability data association, and multiple hypothesis.[1] The distinguishing characteristic separating these approaches is the manner in which data association decisions are made. We will also briefly mention some exciting advanced concepts: Fuzzy Sets and Conditional Event Algebras.

2.1 DATA ASSOCIATION ALGORITHMS

The fundamental multiple sensor, multiple target data association dilemma is the decision on the origin of observations and tracks in the association of observations to observations, observations to tracks, or tracks to tracks. For instance, an observation could be from the individual target of interest, from other targets, from an unresolved CSO, or clutter or false alarm.

There are three fundamental classes of data association algorithms. Assignment algorithms make a definitive decision on the origin of the data at each decision time. No alternative hypotheses are carried into the future to await the assistance of new data in sorting out the truth. Instead of deciding which particular datum to associate, the probabilistic data association (PDA) algorithms average over all feasible associations. Multiple hypotheses algorithms defer a decision on the origin of the data. All viable alternatives are retained as distinct possibilities until later information decides the correct data association. For each class, there are several key concepts, which are summarized in Table S-1 and explained below.

Assignment and multiple hypotheses algorithms have received the most consideration in BMD. In general, assignment data association algorithms should be computationally affordable but may not provide the necessary tracking performance. Multiple hypothesis algorithms, on the other hand, should provide superior performance but their computational requirements may not be affordable.

---

[1] For a more in-depth review see *Survey of Strategic Defense Initiative Algorithms*, Gabriel Frenkel and Barry Fridling, Institute for Defense Analyses, IDA Paper P-2284, November 1989.

Table 2-1. Data Association Algorithms

Type	Feature	Explanation
Assignment	Coordination	Assignment can be performed on each track independently of all other tracks (locally) or on all the tracks simultaneously in a coordinated fashion (globally).
	Dimension	Assignment can be performed on two data lists, such as one set of tracks and the new frame of observations, or more than two data lists such as one set of tracks and multiple frames of data.
	Number	Assignment is usually unique, such as one observation to one track, but can also be multiple, such as multiple observations to one track or vice versa.
Probabilistic Data Association	Dimension	Tracks are updated by an average over all feasibly associated observations from one or more frames of data.
Multiple Hypotheses	Splitting	Create additional tracks for each feasibly associated observation.
	Observation-oriented	Consider each observation in turn as originating from a new target or a feasibly associated existing track.

2.1.1 ASSIGNMENT ALGORITHMS

Nearest neighbor assignment algorithms associate the closest data as calculated by some distance function. For instance, a track would be assigned the nearest observation as calculated by the distance from the track to the measurement.

One approach to single sensor multiple target tracking is to implement a nearest neighbor algorithm for each track independently of all other tracks. This is referred to as uncoordinated or local nearest neighbor. A track would then decide whether to update with an observation independent of how another track updates regardless of competing claims for the observation.

This is unsatisfactory for the reason that associations over multiple targets are interdependent: the association of an observation that simultaneously lands within the gates of more than one track with a particular track denies that observation to the remaining contending tracks. Unique associations are often required in order to ensure statistical independence of the tracks, which considerably simplifies the tracking algorithm mathematics.

Nearest neighbor assignment algorithms should be executed in a coordinated or global manner as follows. A cost matrix is defined for all possible observation to track associations, including that the observations are from a new source (target, clutter, or false signal), and for all track to observation associations, including the case that the correct track observation is not feasible or was not detected. The matrix entries or scores are the probabilities of the associations or their logarithms. In Kalman filter tracking, these are the exponential of minus one-half of the normalized distance between the measurement and track in the association pair. (Other factors can be included in the scores, such as the probability of detection, the probability of finding the measurement associated with the target within the gate, the probability that the observation is from a new source, and the probability of choosing no observation for association with the track.) An algorithm, such as the Munkres algorithm, is executed to assign observations to tracks in a coordinated fashion by maximizing the sum of matrix entries subject to the constraints that no track is updated by more than one observation and one observation is not assigned to more than one track. Maximizing the entries minimizes the total distance between measurements and tracks. The result of such an algorithm is a unique pairing of tracks to observations.

Assignment algorithms usually associate items on two data sets, for instance, a set of observations with a set of tracks. They can also associate two sets of observations or two sets of tracks. Assignment algorithms for two data sets are often described as two-dimensional.

Research and development of assignment algorithms is advancing rapidly. A recent report[2] described an optimal two-dimensional assignment algorithm that is claimed to be substantially faster than even the fastest version of the sparse Munkres algorithm.

Assignment algorithms have been developed that can associate data among more than two data sets, for instance linking several frames worth of observations to tracks. In this manner, multiple dimension assignment algorithms are generating multiple hypotheses in the sense that more than one viable alternative is retained over a number of scans. A hard assignment, that is, decision, is then made after some fixed interval.

---

2 "Comparison of 2-D Assignment Algorithms for Sparse, Rectangular, Floating Point, Cost Matrices," O.E. Drummond, D.A. Castanon, and M.S. Bellovin, in the *Journal of the SDI Panels on Tracking*, the Proceedings of the SDI Panels on Tracking, Issue No. 4/1990, pp. 4-81 to 4-97.

Assignment algorithms are also being developed to make non-unique associations, assigning one track to two observations or two tracks to one observation. These are referred to as multiple assignment algorithms.

Alphatech Corporation has contributed greatly to the development of assignment algorithms, both multi-dimensional and multiple assignment. Alphatech, in part under work funded by the Algorithm Architecture Program of the U.S. Army Strategic Defense Command, has explored the performance of many innovative, advanced assignment algorithms on various computer architectures. This work is of great importance.

2.1.2 Probabilistic Data Association Algorithms

Probabilistic data association (PDA) applies to a single track and is strictly a method for handling the problem of multiple feasible observations with an established track. The fundamental ideas are to average over the latest set of all feasible observations and to utilize the association probabilities of the observations to the track.

In the PDA, each feasibly associated observation is considered as originating with the target. Also the case that the observation originating with the target is not detected or is not considered feasible is given consideration. An association hypothesis is constructed consisting of associating one observation (or none) to the track and considering all others as statistically independent clutter or false alarms.

The PDA procedure is first to multiply the probability of each association hypothesis with the updated state estimate that assumes that hypothesis is true. Then a composite PDA state estimate is formed as the sum over each of these products. This forms a weighted average of the state estimates for each feasible data association hypothesis.

The joint probabilistic data association (JPDA) algorithm extends the PDA to multiple targets by computing the association probabilities jointly across tracks rather than for each individual track. The state estimate is calculated as before as an average over the state estimates for each association hypotheses weighted by the probability for that hypothesis.

The four principal distinguishing characteristics of the PDA approaches are the assignment of many observations to one track, the exploitation of association probabilities, the calculation of state estimates as averages over association hypotheses, and a lack of organic track initiation logic.

The association probabilities are calculated with Bayes Theorem from Probability Theory. For this reason, PDA algorithms are one member of a class of tracking approaches referred to as Bayesian Tracking Algorithms. An Optimal Bayesian Tracking algorithm would extend the association hypotheses over many frames rather than just the most recent.

Bar-Shalom and Fortmann[3] have extended the PDA single scan algorithm to an optimal Bayesian algorithm. Consider a time sequence of observations, one observation from each frame from the initial to the present time. Such a sequence forms one possible target history, that is, one possible track. Consider all possible such sequences. The set of all possible associations at the current scan can be decomposed into tracks at the previous frame associated with some observation from the current frame.

An association probability for each observation sequence, that is, a probability for each track, can be calculated conditioned on the entire set of observations. As in PDA, the conditional probability for each hypothesis multiplied by the state estimate that assumes that hypothesis is true is summed over all possible hypotheses. Thus, the updated state estimate for a track is an average over the different possible association hypotheses.

Since optimal Bayesian algorithms associate over all scans, not just the most recent, the computational expense is prohibitive. A suboptimal approach looks back N frames, referred to as N-backscan, rather than all the way to the initial frame. The original PDA is the zero-backscan suboptimal version.

2.1.3 Multiple Hypotheses Algorithms

An intuitive approach to managing multiple feasible observation to track associations is to split the original track into many tracks, one for each feasibly associated observation. This process is known as track splitting. Each track is updated with the associated observation and carried forward to the next scan in the standard fashion. In this manner, difficult association decisions are deferred until more information becomes available.

A track splitting algorithm has two limitations. First, it has no organic track initiation logic: Observations not feasibly associated with existing tracks are not addressed. The second and most significant limitation of the track splitting algorithm is that

[3] Yaakov Bar-Shalom and Thomas E. Fortmann, *Tracking and Data Association*, Academic Press, Inc., Orlando, Florida (1988).

observation association with multiple tracks is performed in an uncoordinated manner. There is no conflict resolution logic that manages the problem of one observation that may be feasibly associated with multiple tracks.

Deferring difficult data association decisions is archetypical of multiple hypotheses algorithms. In contradistinction to track splitting, other multiple hypotheses algorithms tend to make use of association probabilities like those in the PDA algorithms.

Reid's[4] multiple hypothesis algorithm remedies the absence of organic track initiation logic by generating observation-oriented hypotheses. A track-oriented hypothesis is where every observation is considered for association with each track from the previous scan. No observation is considered for association with a track that did not exist on the previous scan, that is, a new target. This is the reason for the absence of organic track initation logic. In observation-oriented hypotheses, each observation is considered as clutter or a false alarm, as a feasible continuation of a previous track, or as a new target.

Consider the hypotheses generated on the previous frame and the first observation of the new frame. Generate a new hypothesis for each feasible association of this observation: as clutter or a false alarm, as a feasible continuation of a previous track, or as a new target. Take this new set of hypotheses and repeat this procedure with the second observation except that more than one observation cannot be associated to one track. Continue in this way until every current observation has been associated. Reid referred to these as cluster hypotheses.

While the total number of cluster hypotheses generated can be quite large, the number of track-to-observation associations is relatively few, equal to the sum over each track of the number of observations with which it is feasibly associated plus one. This point is important in reducing the number of computations. One track-to-observation association can appear in many different cluster hypotheses. Each association decision is followed by a tracking filter update computation. If the track update computations were performed for each cluster hypothesis then the same filter update computation would be repeated many times. Instead, tracking filter update computations are performed for each feasible association, association probabilities are calculated over alternative hypotheses, and then these are mapped onto the larger set of cluster hypotheses. It must be reemphasized

---

4 Donald B. Reid, "An Algorithm for Tracking Multiple Targets," *IEEE Trans. Auto. Control*, Vol. AC-24, No. 6, December 1979, pp. 843-854.

that each association hypothesis assumes unique observation-track associations so that the association probabilities are calculated over statistically independent states.

The optimal implementation of Reid's algorithm would require ever-increasing computer memory as more hypotheses are generated on each frame. All practical versions limit the number of hypotheses. One way to do this is to divide the set of tracks and observations into independent groups requiring conflict resolution. The growth of hypotheses is also limited by the operations of pruning and merging. Hypotheses considered unlikely, say those below some threshold, are dropped while those that are "similar" according to some criteria are combined. These operations are suggestive of track splitting but in that case there were no association probabilities and there were multiple assignments of tracks to observations.

One limitation of Reid's algorithm is that it does not include multiple associations of tracks to observations, such as may occur in merged measurements, or multiple associations of observations to tracks, such as may occur in track spawning. The fundamental reason for this is the manner in which the association probabilities are calculated. One hypothesis consists of a set of unique associations. The probability of the association hypothesis decomposes into the products of probabilities for the individual components when the states are statistically independent as ensured by unique associations.

Kovacich of Lockheed Missiles and Space Company has described a Bayesian multiple hypotheses tracking algorithm that remedies these defects in Reid's approach.[5] The key idea is to use a Bayesian network architecture (also known as influence diagrams) to provide a calculus to represent and manipulate joint probability distributions such as those that occur in multiple target tracking. Rather than decompose the association-to-track problem into unique association hypotheses, the fundamental unit in Lockheed's approach is the scene which is defined as the joint set of observation-oriented hypotheses, track-oriented hypotheses, and track spawning outcomes for different clusters. The probability for each individual possible outcome is calculated by the Bayesian network. This research is of great importance.

[5] Michael Kovacich, "Application of Bayesian Networks to Midcourse Multi-Target Tracking," *Proceedings of the SDI Panels in Tracking*, Issue No. 4/1989, pp. 4-56 to 4-143.

2.2 ADVANCED CONCEPTS

Fuzzy Sets and Conditional Event Algebras are two of the most interesting advanced concepts under investigation for data association and tracking problems. Fuzzy Set Theory is a significant departure from standard probability theory that could be applied to problems in decisions (data association) and estimation (tracking filters). The fundamental idea is to generalize the classical or crisp set in which membership is dichotomous: an element either is or is not a member of the set. Fuzzy sets eliminate the sharp boundary dividing members and nonmembers by assigning to each element a value or grade corresponding to the degree of membership in the set.

There is a great deal of controversy surrounding Fuzzy Set Theory, Fuzzy Logic, and Fuzzy Control, including:

- Is it good mathematics?
- If it is good mathematics, then does it lead to new insights or methods?
- If it is good mathematics, then is it necessary to master Fuzzy Set formalism to derive the new insights or methods?[6]

Most of the interesting applications to date have been in Fuzzy Control and Fuzzy Signal Processing. We are closely monitoring the application of Fuzzy Set Theory to data association and tracking problems.

I.R. Goodman, of the Naval Ocean Systems Center, and collaborators have developed a new approach to the data association and tracking problem known as conditional event algebras. Essentially, the idea is to create new mathematics for manipulating conditioned random variables and processes. The goal is to develop better mathematics for reasoning based on evidence which would be more appropriate to data association and tracking problems.

---

6 For one view of all this see "Bayesian vs. Fuzzy Theory," Fred Daum, *Proceedings of the SDI Panels on Tracking*, No. 1, 1991.

3. SURVEY OF RECENT TRACKING SIMULATIONS

This chapter surveys three recent tracking simulations that represent the most impressive efforts to date in the community that have been published and with which we are familiar enough to be able to evaluate. For reasons that include competitiveness, not all simulation results are published. These simulations are representative of the simulations that have been published and are representative of the state of the practice in the SDI tracking community.

Observation density is also the single most defining characteristic in SDI tracking simulations, particularly Phase 1 simulations. Tracking simulations measure computational performance (speed and memory) and tracking performance (estimation and association performance measures such as estimation accuracy, credibility, and reliability and track purity). As a practical matter, many simulations emphasize one aspect over the other. Simulations with larger and more dense threats are more often used to investigate computational performance because the data association algorithm and tracking filter design, implementation, and experimentation are compromised by insufficient computational resources. Those data association algorithms and tracking filters are selected more for their computational thriftiness than for their tracking performance. High observation density indicates the level of ambitiousness and complexity in the simulation, determines the adequacy of computer resources, and must be considered in the measurement of the level of difficulty and in the judgement of the quality of tracking performance.

When one considers tracking simulations four issues should be kept in mind:

- *How challenging is the threat being tracked?* Tracking simulations can be likened to diving and ice skating competitions, for instance, in the sense that performance must be evaluated in part by the level of difficulty of the effort.

- *What are the important assumptions and initial conditions?* This is something of an extension to assessing the level of challenge of the threat. There are several key concerns: To what extent are clutter and background effects included and how are they modeled? What is assumed for sensor properties such as resolution and accuracy? How is the sensor and signal processing modeled? Is sensor tasking scripted in advance or performed on-line? Are data misassociations permitted? Start up conditions can grossly affect tracking performance. For midcourse simulations the key issue concerns how midcourse tracks are initiated: Does the simulation rely on tracks handed over from boost phase and if so how good are those track hand-overs?

- *What are the key algorithm details?* It is important to know some detail of the algorithms and algorithm architecture to appreciate the context of the work.

- *What are the scoring methods and measures of effectiveness?* How are tracks associated with true targets for evaluation?

3.1 ADVANCED SYSTEMS ARCHITECTURES

Advanced Systems Architectures of the United Kingdom, under contract to SDIO, has investigated the performance of their algorithm against the unclassified 460 object Test Case 1 of the SDI Panels on Tracking. Their algorithm, referred to as The Target-Oriented Approach to Data Fusion, was described in detail in last year's survey. This description of their work is based on an ASA report[1] and on a presentation to the Panels, which can be found in the Proceedings.[2]

They considered tracks for only those 380 objects which were continuously visible throughout the scenario. One track for each of these 380 objects was initiated by very optimistic hand-over information: one second before the scenario begins, diagonal covariance matrices are formed with 7 meters position error and .025 meters/second velocity error. Cluster tracks for centroid and extent were initiated with the values corresponding to the cluster's true members individual object track values. We consider this to be unrealistic.

A key algorithm detail is the philosophy of their approach, which is to perform individual object and cluster tracking with processes operating as concurrently as possible. This was a significant factor in the selection, implementation, and investigation of the data association algorithm. A global nearest neighbor assignment of observations to tracks was rejected in favor of a local nearest neighbor assignment in order to maintain insofar as possible the concurrence of the track processes. It was unclear from the information

---

1 *Simulation and Demonstration of the Target Oriented Sensor Data Fusion/Tracking Algorithm for the SDI Mid-Course, Final Report*, ASA REF: T90/007, 30 September 1990, Edward Goodchild.

2 *Proceedings of the SDI Panels on Tracking*, Issue No. 4/1990, pp. 1-131 to 1-167.

available to us what was done with extra observations, if there were any from one frame's data, that were feasibly associated with a track.

Track to truth association was accomplished via a target-oriented local nearest neighbor assignment: For each true target state considered independently of all others, the track nearest to it was assigned for purposes of evaluation. The measures of performance were the magnitude of position error and velocity error, for both the individual object tracks and the cluster tracks.

Various computers were used to investigate different computer architectures, including Alliant FX 80, Alliant FX 2800, MIPS M120-5, and a Cray Y-MP.

3.2 ALPHATECH

Alphatech in the Algorithm Architecture Program[3] analyzed portions of the unclassified Test Case 2 of the SDI Panels on Tracking consisting of 750 objects and portions of an unclassified 6,260 object threat case provided by Albert Perrella from IDA using the SAAM software package. Key assumptions and start-up conditions were 20 microradian sensor measurement accuracy and 100 microradian sensor resolution and hand-over track accuracy of either 200 meters and 20 meters/second on some objects in the threat, not necessarily the booster or post-boost vehicle, or 350 meters and 4 meters/second. We consider these start-up conditions to be unrealistic. Alphatech is moving, however, to incorporate into their simulations cold start initial orbit determination algorithms. The computers used were SOLBOURNE, DAP 510 & 610, and ALLIANT FX/8.

In one simulation, 50 objects from the 750 object case were used to compare multi-dimensional assignment algorithms, including:

- Multi-dimensional maximal marginal return (M^3R)
- Backtracking (from AT&T)
- Branch and bound.

In a second simulation, the threat was 125 objects from the unclassified 750 object case, consisting of one booster delivering one post-boost vehicle, with three reentry

[3] *Algorithm Architecture Program Subsystem Requirements Review*, 22-23 August 1990, Prepared for the Department of the Army, U.S. Army Strategic Defense Command, and Interim Progress Review, 15-16 May 1991.

vehicles, and six cannisters dispensing 114 balloons. The goal was to compare the tracking performance of two different assignment algorithms: maximal marginal return (MMR), which is a two-dimensional version of M^3R, versus multiple assignment. In the MMR case, unassigned observations were used to spawn new tracks using the nearest tracks, in a version of track splitting. One measure of performance was average three-dimensional position error as determined by unique assignment of tracks to targets based on minimizing the sum of three-dimensional position errors.

A third simulation, driven by the portions of the unclassified 6,260 object case, experimented with various computer architectures and parallelization. The measure of performance was computer processing time. The goal was to process one frame's worth of data in 10 seconds or less, which is considered one nominal frame time. No tracking accuracy performance scores were considered.

3.3 FULL-THREAT CLOSED-LOOP SURVEILLANCE ALGORITHM EXPERIMENTS AT THE NATIONAL TESTBED

Larry Stalla of the National Testbed (NTB) Integration Contracting Team conducted the most ambitious and impressive SDI tracking simulation to date with which we are familiar. His results were briefed to the SDI Panels on Tracking and, absent the classified data, can be found in the Proceedings.[4] Stalla used the test environment and test articles integrated into the Version 2.3 Simulator by the NTB Integration Contractor at NTB Joint Program Office--Simulation Directorate (NTBJPO/SD) direction between September 1988 and November 1990. The primary objective of Stalla's experiment was to demonstrate a capability to simulate a launch-to-impact scenario, so called end-to-end, using a complete suite of test articles in a simulation environment of realistic full-scale threats and sensor performance. A secondary objective was to characterize the performance of the test articles.

Stalla conducted an experiment for each of four levels of classified threat: the full design-to-threat DTT-1; the full phase-one threat scenario POTS-3A; the DTT-1 threat with decoys removed; and the DTT-1 threat with decoys removed and perfect sensor resolution assumed. Each experiment was run three times:

4 *Proceedings of the SDI Panels on Tracking*, Issue No. 1/1991, pp. 3-123 to 3-159.

- The baseline run in which the full set of data association, tracking (state estimation or filtering), and sensor tasking algorithms were implemented as an integrated suite, in a "closed-loop;"
- A run in which the data association algorithms were not implemented, instead data association was performed essentially perfectly based on true object identification; and
- A run in which in addition to perfect data association, tracking was performed using true target state information, leaving sensor tasking as the only actual test article.

Because of the scale and scope of the experiments, we will not in this limited space attempt to describe the details of the algorithms that were implemented as test articles. Suffice it to say that cold and warm start track initiation algorithms and track maintenance algorithms were integrated. Hand-overs between sensor elements were simulated, including boost-phase to midcourse (BSTS to SSTS), space-based midcourse sensors to ground-based probes (SSTS to GSTS) launched based on the information in the run (both track estimates and target truths). Various assignment algorithms apparently were used for data association.

Stalla's measures of performance included the number of kills and misses by the Ground-Based Interceptor (GBI); the fraction of total GBI divert capacity used per divert; unnormalized track position error distribution for each sensor system; and the constellation coverage efficiency in terms of the number of object sighting messages (OSMs) reported from each sensor platform. There was also a score for data association accuracy. Classification limitations prevent discussion of the results here.

Stalla's experiment has generated controversy, mostly due to the poor performance exhibited by the test articles. We were quite positively impressed by the scale and scope of the experiment, which no other facility in the country could even have attempted. This reflects most favorably on the NTB. The performance of the test articles was indeed poor, which some have construed as reflecting poorly on the NTB. We feel this criticism is not justified. Our understanding is that the Version 2.3 Simulator is not an algorithm design program but rather an algorithm integration program. The test article performance was exactly a reflection of the quality *and* the state of development of the algorithms that were integrated. SDI, and particularly Phase 1, surveillance, tracking, discrimination, battle management, and interceptor guidance are the most challenging problem in defense research and development today. Therefore, it should be a surprise to no one that a test article integration program that precedes vigorous algorithm experimentation would end up with test articles that perform poorly. Stalla's experiments are not representative of what tracking performance is eventually possible and should not be construed in that manner.

3.4 CONCLUSIONS

Good performance of data association, data filtering, data management, guidance, discrimination, etc., cannot be assured by algorithms anywhere in the community, otherwise SDI Phase 1 surveillance, tracking, battle management, and interceptor guidance would be a "solved problem." It most certainly is not that. Many experiments focused specifically on test article development will be required to begin to understand in detail the keys to good performance and bad. Progress towards "solving the problem" will be made only by widely disseminating the methods and results of experiments throughout the SDI community. Test article experimentation is the primary justification for the existence of the STB separate from the NTB. It would be an error of major proportion for there to be a controversy surrounding poor results that might inhibit the critical flow of information and results from SDI simulation experiments.

4. CLUSTER TRACKING

One approach to managing the high density SDI threat, particularly during early midcourse, is to forgo tracking individual objects and instead to track closely spaced individual objects as a group or cluster. Following the Panels, we will distinguish between targets as they are in truth, referred to as a group, and targets as they are observed in sensor measurements, referred to as a cluster. Group will also be used generically to refer to both. Since individual target tracks are ultimately what is required, the key issues in cluster tracking are:

- Why do cluster tracking?
- When is cluster tracking performed rather than individual target tracking and vice versa, and when and how is the transition between cluster and individual object tracking accomplished?
- What type of cluster tracking is performed?

The Panels have defined a spectrum of group tracking approaches:

- *Group:* Group properties alone are tracked.
- *Group with Simple Individual:* Simple individual object information is tracked but the tracking of group properties is emphasized.
- *Individual and Simple Group:* Simple group information is tracked but individual object tracking is emphasized.
- *Individual object tracking:* Individual object tracking alone is performed.

The Panels have also identified a set of cluster tracking algorithm architectures. These can be found in the previous survey.[1]

4.1 INTRODUCTION

Shortly after deployment from post-boost vehicles (PBVs), reentry vehicles (RVs) and decoys are so closely spaced that sensor observations consist of unresolved clumps of objects and extended objects. A clump is an observation arising from two or more targets

---

1 *Proceedings of the SDI Panels on Tracking*, Issue No. 1/1991, pp. 3-123 to 3-159.

that appears to be from an individual object, as would happen when two or more targets are not resolved by a sensor. An extended object is an observation extending over many more pixels than an observation from a single object. With increasing time from deployment, clumps may resolve into individual observations and resolved closely spaced observations may spread as the targets disperse.[2] Target resolution is a function of the optical sensor resolution *and* the viewing geometry and range. For this reason, as sensors move along their orbits, observations could resolve or unresolve, spread or contract.

It is not possible to establish individual tracks on the targets in a clump since individual object tracking is possible only on individual observations. The density of closely spaced resolved observations may compel cluster tracking over individual object tracking as the only practical alternative because of great computational expenses in track initiation and misassociation. If targets contract or unresolve, the tracking architecture must transition between individual object and cluster tracking.

There are critical operational requirements for maintaining tracks on individual targets, including discrimination of RVs from decoys, threat assessment, and threat engagement. Cluster tracking is performed when individual object tracking is impossible, too expensive, or not necessary. As the threat resolves, cluster tracks spawn individual object tracks by initializing individual object tracks from the cluster track. Cluster tracking should be evaluated based on relatively inexpensive computation and communication requirements and the quality of the initial estimates for the spawned individual object tracks.

4.2 CLUSTER TRACKING

For resolved closely spaced observations a cluster track develops estimates on some group properties, such as cluster centroid position and velocity and centroid extent. A cluster gating logic that is a generalization of that for individual object tracking determines the observations to be considered for updating the cluster tracks. A conflict resolution logic is required for all observations that satisfy multiple cluster track gates. All observations assigned to a cluster track are used to compute the measurement centroid and possibly the measurement dispersion. The measurement centroid updates the cluster centroid state in the standard manner of Kalman filtering. The modeling of the dynamics of the cluster extent

2 This is not to suggest that the threat density will not or cannot be increased later in the flight.

distinguishes most approaches. Tracks for objects splitting off the cluster are initialized by the cluster centroid state.

Drummond, Blackman, and Hell[3] have extended cluster tracking to multiple sensor cluster tracking, where the principal difficulty is that the observed size and shape of the cluster varies from sensor to sensor. For this reason, multiple sensor cluster tracking must have more information than just the location of the cluster. Drummond et al.'s approach is to model the group as an ellipsoid in three dimensions. Separate filters are established for the group centroid and the ellipsoid extent parameters. The group centroid state estimate initializes the tracks for objects that split away from the group, as before. The ellipsoid extent state estimate permits sensors in different positions to associate groups.

The cluster tracking efforts with which we have some detailed information are described in Table 4-1.

Table 4-1. Cluster Tracking Algorithms

Contractor	Principal Developers	Brief Description
Hughes Aircraft and General Dynamics	Oliver Drummond, Samuel Blackman	Multiple Sensor algorithm for tracking cluster centroid and extent.
MIT/Lincoln Laboratory	C.B. Chang, Keh-Ping Dunn, Ming Tsai	Individual tracks for the observations defining the edge of a cluster to form tracks for cluster members, which are assumed to be moving in parallel with edges.
Nichols Research Corporation	Robert Osterheld, Lisa Brahm	Develop individual object tracks of RVs and decoys deployed off a common PBV through their closely spaced phase.
Teledyne Brown	Keith Maples	Track cluster centroid and the parameters of an RV-decoy deployment model to develop individual object tracks.

3 O.E. Drummond, S.S. Blackman, K.C. Hell, "Multiple Sensor Tracking of Clusters and Extended Objects," *Technical Proceedings 1988 Tri-Service Data Fusion Symposium*, Laurel, Maryland, May 1988.

4.3 CONCLUSIONS

Much more work needs to be done to explore the diversity of algorithms and algorithm architectures and the critical issues associated with cluster tracking. Work to date has only begun to address the problems and possibilites.

The Surveillance Testbed (STB) will provide an important environment to investigate these issues. There is one cluster tracking algorithm in the initial set of test articles being hosted on the STB. The status of cluster tracking remains for the most part as it was last year: In need of experiments and analyses that explore the diversity of cluster tracking algorithms and algorithm architectures and the critical issues outlined above.

5. SURVEILLANCE TESTBED (STB)[1]

Substantial progress has been made towards construction of a testbed to experiment on and evaluate alternative surveillance algorithms--detection, tracking, discrimination, and sensor tasking--and to assess ballistic missile defense system-level surveillance performance. While the STB is being constructed to support both purposes, to the extent that one takes precedence over the other, first priority must go to evaluation of surveillance algorithms: An emphasis in favor of experimentation on algorithms increases the prospects for a testbed that successfully assists in the development of surveillance algorithms and accurately assesses system-level surveillance performance. The program seems on track to ensuring the creation of a facility that is sorely needed by the SDI tracking community.

The Surveillance Testbed is critical to SDIO. The STB will be the SDIO facility where contractors can with a minimum of modification run their own software in high fidelity, high detail surveillance simulations to verify detection, tracking, discrimination, and sensor tasking. A contractor's algorithms can be tested by themselves on the STB as well as in their role in a complete surveillance system by inserting them into a complementary set of "house algorithms," algorithms maintained on the STB. With SDIO approved threat scenarios, and SDIO supplied signal generation and sensor and signal processing data that are the inputs to the surveillance algorithms, critical algorithm experimentation and evaluation can take place in a controlled environment and scored according to standardized methods, thereby facilitating performance comparisons. By having the STB supply the inputs that drive high fidelity, high detail simulations, SDIO saves the resources otherwise spent when each contractor develops their own extensive simulation drivers.

---

1 This section is based on material from briefings given by Mike Wesley of Nichols Research Corporation (Huntsville), to the SDI Panels on Tracking, from the 15 October 1990 Coordination Draft of the Surveillance Test Bed (STB) Build 1 (Phase 1) Design Documentation, CDRL 119, from the STB Build 1 Functional Requirements Document, CDRL A099, 1 February 1991, and CDRL A099-1, 1 July 1991, and from meetings of the STB Test Article Interface Working Group.

5.1 THE STB DESIGN

The STB is comprised of the Test Environment, which is the fixed experimental facility, and Test Articles, which constitute the subjects of the individual experiments. The STB eventually will consist of simulations to experiment on and evaluate alternative surveillance algorithms, which are used to develop and validate functional representations of, or data bases for, surveillance algorithms. Based on these functional representations and data bases, large-scale simulations will quantify system-level performance of the surveillance system.

The Test Environment is comprised of the Test Driver and the Framework. It is within the Driver where the user specifies threat scenarios and sensor suites and where threat modeling, environments modeling, scene generation, and sensor/signal processing occurs. Trajectory propagation of the threat and sensors will also be done by the Driver. The outputs of the Driver will be sensor measurements: object sighting messages (OSMs) and radar pulse returns for the sensors. User control of the simulation set-up, execution, monitoring, and output format are controlled by the Framework.

A typical STB experiment will have a definition and pre-processing phase, an execution phase, and a post-processing phase. Construction of the inputs to the test articles, that is, the test drivers, will be performed in the definition and pre-processing phase, beginning with the specification of a threat from a master threat tape. For each of the objects in the threat, across their entire trajectories, high detail target signature data bases are constructed to the extent possible from SDIO's standard phenomenology code: the Naval Research Laboratory's (NRL's) Strategic Scene Generation Model (SSGM). These trajectory and signature data bases are constructed without reference to a specific surveillance sensor architecture. To this point, test drivers can be standardized for all experiments.

Each particular experiment is specified by a sensor architecture that includes details on sensor orbits, sensor fields of view, and various sensor models, parameters, and signal processing algorithms. Target trajectory and signature data bases, together with high detail background signature data bases, are inputs to sensor focal plane and signal processing models to generate sensor measurements. The sensor measurements that result, which are referred to as test drivers, are the inputs to surveillance algorithms. These test drivers can be standardized only to the extent that the sensor details and the sensor fields of view are.

The STB is to be constructed in two distinct phases, the so-called Build I and Build II. According to the 1 February 1991 Functional Requirements Document,[2]

> Build I will concentrate on the development of a simulation framework that will provide the necessary drivers and executive functions to control the simulation and feed the STB test articles with data necessary for their operation. The Build I test article algorithms will come from existing programs or from element development programs. Build I will provide an initial operational capability (IOC) testbed for evaluating SDS [the Phase 1 Strategic Defense System] surveillance requirements and resolving critical feasibility issues in an evolutionary fashion with intermediate capabilities supporting ongoing analyses. Build II will follow Build I development and will provide a fully modular testbed for evaluating alternative element surveillance algorithms, models, and concepts of operations at a system level.
>
> The purpose of the Build I STB is to provide a simulation capability to investigate the SDS surveillance functions and performance capabilities for the SEIC [Systems Engineer and Integration Contractor], the SDIO, and other designated government and contractor analysts... . The purpose of the Build II STB is to provide a fully robust test environment to evaluate alternative surveillance algorithms or design implementations for the SDIO, the SEIC, and other designated government and contractor analysts.

5.2 STB ACTIVITY OF THE SDI PANELS ON TRACKING

The SDI Tracking Panels have played a critical role in the development of the STB. Before the STB program was started the members of the Tracking Panels, representing tracking algorithm designers from across government, FFRDCs, contractors, and academe, called for an algorithm testbed of this sort. A testbed on which experimentation could be performed and high detail, high fidelity threat scenarios complete with sensor measurement data could be produced and supplied to contractors for use in their own testbeds. During the development of the STB, the Panels provided a peer review of the STB development plans and supplied the input of the intended user community. The Panels deserve credit for helping to shape the STB into a facility that should well serve SDIO and its contractor community. In this section, we review the Panels' role in the STB.

The Panels raised several critical issues when the initial STB development plan was presented. In a resolution delivered to SDIO, the SDI Panels on Tracking during the 27-29 November 1990 meeting held that:

---

2 Op. cit., Ref. 1, pp. 1-4 - 1-5 (see p. 5-1).

1. The value of the Surveillance Testbed (STB) to the Strategic Defense Initiative Organization will be greatly enhanced by constructing the Build 1 STB to be flexible enough to accommodate the diversity of tracking algorithms. Among other things, this requires an emphasis on the interfaces between test articles and test environment, including feedback from the tracker to the signal processor. The Panels will work with the STB to detail these interfaces. The Panels urge that preparations for accommodating alternative algorithms be a principal requirement of the Build 1. The Panels are very concerned that in the absence of this requirement in the Build 1, STB will fail to be a tool for developing and demonstrating tracking algorithms and STB will fail to quantify strategic defense system performance accurately.
2. Early conclusions on system level performance and system level requirements drawn from experiments using baseline algorithms may be of limited validity and should be viewed with reservation. Baseline algorithms need not be representative of the performance to be achieved by alternative algorithms. Program offices should have a mechanism to comment on the data and results produced by the STB.

The Panels' recommendations to the STB are summarized as:

- Development of flexible interfaces between STB test environment and tracking algorithms must be emphasized early in order to best ensure that the STB achieves its goal of providing a testbed for developing and evaluating alternative surveillance algorithms;
- Limited emphasis should be given to early results of system-level performance experiments with the "representative, baseline" algorithms used for testbed integration validation.
- The STB must support a portable testbed facility.

Interfaces

The Panels identified and analyzed three different critical interface issues for the STB:

- Interfaces between individual test articles and the STB test environment;
- Interface requirements stemming from permitting feedback from tracking algorithm to sensor and signal processor, such as sensor tasking; and
- Interfaces between test articles.

It should be no surprise that hosting tracking algorithms on the STB will require some software modifications. Everyone can agree that it is imperative to keep the modifications to a minimum. The real issues are what sort of modifications, how many, and by whom, the STB contractor or the tracking algorithm contractor? For the most part, it must be the tracking algorithm developer who modifies their software since they are the most knowledgeable of the code's contents. But their willingness to utilize the STB depends on the scope of the modifications. The STB's government sponsors need to appreciate that, everything else in order, the STB will fail or succeed based on the scope of software changes required for hosting surveillance algorithms.

To appreciate interface demands on tracking algorithm software, it is necessary to describe the software structure of the STB. Within the STB, data are maintained in logically related groups known as data objects. For example, there are data objects for object sighting messages (sensor measurements) and for tracks.

Tracking algorithms will be implemented as subroutines within an Ada program that is referred to as an Ada shell. The shell unloads data objects into the data structures corresponding to the tracking algorithm argument list via devices called ports and parameters. The shell calls the tracking algorithm, moves the data in, and returns the data to the system data objects after completion of the tracking algorithm.

Tracking algorithm-data object interface is provided by ports and parameters. Ports (to the data) permit the tracking algorithms to access the data objects and get the data. Parameters permit tracking algorithms to access the data applicable only to it, for instance, preventing the tracking algorithms for accessing truth, which is maintained in the data objects for post-processing performance evaluations. Parameters also adapt the data in data objects to the specific needs of the tracking algorithms. The ports and parameters are key devices that permit the interface of multi-party software not written to be interfaced.

The use of ports and parameters does not preclude data defined internally within a tracking algorithm, for instance variables for intermediate results in computations. But, all internal data not passed out to data objects through the ports and parameters will be lost, that is, will not be saved between calls of the tracking algorithm. The reason for this is essentially that the STB software simulates a constellation or system of sensors not by replicating the surveillance algorithms, one complete set for each sensor, but rather by maintaining data objects for each sensor and using one common set of surveillance algorithms. To execute a particular sensor's surveillance function, the STB's simulation executive inputs that sensors data objects into the common set of surveillance algorithms.

Between succeeding executions of a particular sensor's surveillance function, another sensor's call to the common set of surveillance algorithms will overwrite the internal data.

For internal tracking algorithm data that need to be retained over multiple calls, the tracking algorithm, its Ada shell, and the STB data object system must be modified to establish system data objects for storage of these additional data and ports and parameters to permit access. Potential users need to analyze STB data objects and their own tracking algorithms to identify those internal data that need to be stored in additional STB data objects. The STB needs to prepare to accommodate those additional data objects and to modify the ports and parameters accordingly.

The Panels were concerned that the STB accommodate the many tracking algorithms that are designed to send information from the tracker to the sensor and signal processor to affect their operations. For instance, sensor tasking algorithms determine sensor fields-of-view based on tracking information. These algorithms are critical to the performance of some surveillance algorithms, particularly for sensors with narrow fields of view. Another example is where tracking information is used to vary the sensor scan pattern or to vary detection thresholds across the field of view.

The STB now includes sensor tasking algorithms as test articles in high detail simulations. This means, however, that many of the tasks that were to be done in the pre-processing phase will now have to be done on-line during the simulation as tracking information is used by the sensor tasker to determine fields of view. Background clutter scenes will have to be produced on-line during the simulation. Target detection and signal processing will also have to be done on-line rather than in advance since what the sensor sees will not be determined in advance of the simulation.

Tracking algorithm functions that are considered by the STB as basic test article units are:

- (Cold start) Track initiation
- Track continuation
- Differential initialization (warm start/track splitting)
- Track fusion
- Cluster tracking
- Sensor tasking.

Any subfunction within these, such as a Kalman or least squares filter or data association algorithm, cannot be tested except by being placed within one of the above basic units. An experimenter's tracking algorithm could consist of any number of basic tracking algorithm functions. A "house set" of tracking algorithm functions will be maintained on the STB available to supplement those functions within an experimenter's tracking algorithm.

Interfaces between tracking algorithm functions are, in general, highly non-trivial. After all, these interfaces manifest the logic of the tracking algorithm. Tracking algorithm interfaces can and will vary widely with algorithm and with contractor. Large contractor efforts will probably have a mostly complete set of tracking algorithm functions and will probably not have any serious tracking algorithm to tracking algorithm interface issues. Small contractor efforts, on the other hand, may not have a complete set of tracking algorithm functions and may be looking to the STB for supplementary functions. In this case there will be a difficult job of interfacing those functions.

Early System-Level Evaluation

The STB is not an algorithm design program. In the Build I, "representative, baseline" surveillance algorithms readily available will be hosted so that integration testing of the simulation framework, test drivers, and test articles can be performed. Baseline surveillance algorithms used by the STB contractor for integration testing, however, may not be representative of the performance of surveillance algorithms being developed by other SDI contractors. Conclusions on system-level performance drawn from experiments using baseline algorithms that are not representative in terms of performance would be of limited validity or even misleading. A redefinition of surveillance requirements as a result of such experiments would be a mistake.

Having the means to evaluate alternative surveillance algorithms using high detail test drivers is of the utmost importance to SDIO. Creation of such a facility is extremely challenging and ambitious in its own right. Only after a host of algorithms have been implemented can performance which is representative be appreciated. With competitive algorithms in hand, the STB will be in position to quantify strategic defense system performance accurately.

Portable Testbed

A portable testbed facility can be created either by distributing computer threat tapes or permitting the operation of tracking algorithms from remote hosts. There are several test tape formats. Usually sensor tasking and signal processing are done in advance so that a threat tape would consist of sensor orbits and true target states (including intensity) and measurements. To permit the incorporation of sensor tasking, a test tape will have to be accompanied by a rudimentary signal processing algorithm to generate sensor measurements on-line, during the simulation.

The STB has recently decided to permit the capability of running tracking algorithms on the STB from remote sites. This is most welcome and important. In fact, for tracking algorithm experimentation, this may turn out to be the dominant mode in which the STB is used initially. Tracking simulations running from a developer's own computers could use the STB to generate sensor measurements on-line during a simulation in response to calls from the tracking algorithms running on the remote host. In this mode, the STB would essentially play the role of sensor.

Permitting remote host operation of tracking algorithms on the STB usefully segments the STB user community into three groups:

- *Low-cost algorithm experimentation:* threat tapes could be distributed with sensor architecture, true target states, sensor measurements, and rudimentary signal processing algorithms.

- *High-fidelity algorithm experimentation:* remote host operation of the STB to gain its sensor capabilities, including sensor tasking; no access to STB surveillance algorithms.

- *Algorithm evaluation and access to STB surveillance algorithms:* hosting of algorithms on STB; integration of algorithms in surveillance system to demonstrate performance; government evaluation of surveillance algorithms in a controlled environment and scored according to standardized methods to facilitate performance comparisons.

The attractiveness of this segmentation is that the STB can be all things to all users. Flexible high detail and fidelity algorithm experimentation can be conducted by contractors from their own facilities at a minimum cost without going through the rigors of hosting algorithms on the STB. After successful development work, these contractors would be very motivated to modify their software as needed to permit the government to evaluate their algorithms on the STB.

5.3 CONCLUSIONS

Testbed priorities must be clearly established:

- Vigorous surveillance algorithm development must precede surveillance system evaluation;
- Prepare early to accommodate the diversity of tracking algorithms and algorithm architectures;
- Interfaces between test environment and tracking algorithms must be flexible and robust;
- Only after representative surveillance performance is appreciated and quantified by experiments on individual surveillance algorithms can system-level surveillance performance be assessed accurately.

The SDI Panels on Tracking will continue to work with the STB, to provide the SDI tracking community and algorithm experts a voice in the construction and operation of the STB.

PART II

SOME REMAINING PROBLEMS PERTAINING TO THE DESIGN, UTILIZATION, AND EVALUATION OF TRACKING ALGORITHMS

6. SYSTEMS ISSUES IN MULTIPLE SENSOR BMD TRACKING

6.1 BIRTH-TO-DEATH TRACKING

Cues from boost surveillance satellites to midcourse and terminal systems can consist of state vector estimates and estimation error covariance matrices for the states of the boosters at the end of powered flight, known as burnout, or at the times of the satellites' last observation for each booster. These estimates are then predicted ahead to future times, in the first instance by propagation along mostly free-fall orbits, and in the second case by extrapolation or prediction to burnout preceding free-fall propagation. Cues from midcourse passive electrooptical surveillance satellites to ground-based radars can consist of state vector estimates and estimation error covariance matrices for the positions and velocities for the free-falling objects after the post-boost phase, at the times of the satellites' last observations for each of the objects.

In general, trajectories are completely specified for all times by knowledge of the so-called initial conditions. Uncertainty in the initial conditions implies uncertain knowledge of the trajectory later. In addition, if there are dynamical model errors then the trajectories are uncertainly known regardless of the precision with which the initial conditions are specified. For the payloads of ballistic missiles, midcourse orbits are largely determined by position and velocity at the end of booster powered flight. The intermittent accelerations of the post-boost vehicles will alter the orbits and earth impact points of the payloads, as will aerodynamic forces.

Tracking algorithms during the track initiation phase estimate the initial conditions and their error based on some initial set of observational data. Subsequent data are used to refine the estimates of target trajectories and errors during the track maintenance phase.

Track initiation refers to the process of initializing a full state track for the three-dimensional motion of the target. Cold start track initiation is the process of computing an initial state estimate vector and an initial state estimation error covariance matrix from the time sequence of observations in a focal plane track. It is usually comprised of focal plane tracking (also known as track assembly) and initial trajectorv/orbit determination.

The assembly of observations in a focal plane track is considered a candidate full state track. Each focal plane track consists of a time sequence of measurements, one from each of multiple successive frames of sensor data, linked together by the hypothesis that they are from the same target. In focal plane tracking the dynamics of the targets are approximated by simple models. Rather than explicitly modeling the three-dimensional dynamics of the targets' motion, azimuth and elevation angles dynamics are modeled as independent linear or quadratic functions. The Kalman filters that result are referred to as polynomial filters for constant velocity or constant acceleration targets. As an alternative to filters for azimuth and elevation, independent polynomial filters for each component of the line-of-sight vectors can be used to track the targets in the focal plane. These avoid the discontinuities in azimuth as targets pass through the sensor's nadir and zenith points.

Booster track initiation is usually accomplished by the algorithm variously known as template matching or profile matching. Succinctly, templates are historical averages and *a priori* assumptions of booster trajectories derived from data observed by national technical means and trajectory reconstruction programs. There are templates for altitude, ground range, acceleration, mass, intensity, etc., versus time from launch for each missile type and mod. Two fundamental issues, which we address below, immediately attend this use of historical averages and *a priori* assumptions on booster trajectories:

- To which template should the observed data be matched?
- What is the reliability and credibility of relying on templates?

Midcourse angles-only initial orbit determination is an old subject with a pedigree that extends back almost two hundred years to Gauss and Laplace. The fundamental midcourse tracking challenge to this time has been the large number and high density of missile booster, RVs, decoys, and clutter to be tracked. As the severity of the threat declines, as measured by density of objects seen by a sensor, from a defense against a full Soviet attack, to a defense against accidental or unauthorized missile launches, to theater missile defense, data association becomes less of a concern. Single satellite angles-only track initiation in the dense observation environments of Phase 1, and perhaps GPALS, is a critical issue.

The issues in BMD cold start track initiation, both for the template matching in boost-phase and for midcourse, are first the ability, in a dense observation environment, to assemble a reasonably small number of credible time sequences of angles-only measurements without being able to use models for the detailed models for the three-dimensional target motion. The other issue is the reliability of the initial trajectory/orbit determination algorithms and the accuracy, precision, and credibility of their initial state estimates and estimation errors. Low precision state estimation errors will lead to greatly complicated data association problems for track maintenance in dense observation environments. The critical issue is to gain high enough precision estimation errors to mitigate data association problems. We know of no simulations where these issues are thoroughly examined.

In a fully integrated ballistic missile defense surveillance system, tracks would be disseminated and fused throughout the various battle managers and sensor elements: boost-phase surveillance sensors, space-based and ground-based space surveillance sensors, and terminal-phase sensors. Ballistic missile defense birth-to-death tracking is the concept of maintaining continuous tracks on targets from launch through to impact by fusing tracks across sensor elements. Birth-to-death tracking schemes range from the grand to the temperate.

In the grandest design of birth-to-death tracking, downstream narrow field-of-view sensors that lack adequate independent search capability are enabled by pointing directions from upstream sensors. Additionally, downstream sensors initialize tracks by relying entirely on upstream track hand-overs. Boost-phase tracks would be handed over to the post-boost and midcourse surveillance sensors for continuation: booster burnout position and velocity would initialize the post-boost vehicle (PBV) track. Narrow field of view optical midcourse sensors would be directed where to look for the PBV rather than independently searching. Ground-based radars would also be cued where to look for midcourse and reentering objects to gain increased detection ranges. As the PBV dispenses reentry vehicles (RVs) and decoys, tracks for each would be established by spawning new tracks from the continuing PBV track. Every object in the midcourse then could be traced back to its origin, PBV and booster, and a track for each established essentially by continuation of booster tracks. There would never be need during midcourse for the "cold start" track initiation procedures of assembly of a sequence of measurements for initial orbit determination data processing. All midcourse tracks would be initialized by "warm start" track initiation: hand-over of track data from upstream sensor elements and spawning new tracks from existing tracks.

With track hand-overs, the battle manager can launch and guide ground-based interceptors to their targets entirely independent of their ground-based radars. Midcourse tracks could be handed over to interceptors as in-flight target updates to enable on-board guidance algorithms to fly the interceptors into the close proximity of their targets, where the interceptor's own on-board sensors would take over. Midcourse tracks handed over to the battle manager could also permit the early commit of interceptors before their radars see the targets but, in this case, the radars would guide the interceptors as the targets come into view. Ground-based radars could use midcourse track hand-overs to cue their search to concentrate energy into narrow fields of view to gain increased detection ranges. The common goal of these cueing schemes is to enhance the battle space of the ground-based defense system.

Birth-to-death tracking in its grandest design is logical and efficient. It possesses the virtues of omniscient accounting of threat objects, enabling of relatively inexpensive narrow field of view optical sensors, and avoidance of the immense computational expense and complication of cold start track initiation. But it counts on the existence of a seamless link across sensor elements, in which upstream track information is available exactly when and where it is needed. Such a link is ambitious, perhaps even feasible. But a surveillance system that is entirely reliant upon it is critically susceptible to catastrophic failure.

Temperate birth-to-death tracking schemes cue downstream sensors to assist (rather than enable) relatively narrow field of view optical sensors, to increase the battle space of ground-based BMD systems, and to avoid cold start track initiation but only if and when upstream track hand-overs are available. Hand-over tracks are not considered to supplant an independent operational capability for each individual sensor or individual sensor element.

Ultimately, it is a question of capability versus robustness against catastrophic failure. Each sensor element and each individual sensor must be capable of searching a reasonable surveillance region and performing cold start track initiation to reduce vulnerability and to constitute a system that works in a world of communications delays and misdirections and the unexpected, but only if adequate capability is achievable or affordable in individual sensors or sensor elements.

6.2 BOOSTER TRACKING AND TEMPLATE MATCHING

Boost-phase tracking experience to date is limited. What can be performed well is the tracking of a small number of targets across the focal plane (focal plane tracking) of a passive electrooptical sensor on board geostationary satellites without explicitly modeling the boosters' detailed three-dimensional dynamics or explicitly tracking the boosters' three-dimensional trajectory. Historical averages and *a priori* assumptions for booster altitude versus ground range flight profiles and intensity versus time profiles, known as templates, are used to estimate launch locations, times, and azimuths, and to identify missile types. But with regard to using focal plane tracks and templates to predict future states in post-boost, midcourse, and reentry, the accuracy required for BMD has not yet been demonstrated. Brilliant Pebble booster tracking of a particular sort has been demonstrated in limited simulations with low fidelity data and without clutter. Extensive simulations that demonstrate convincingly the tracking accuracy required for booster surveillance, good weapon-to-target assignment, and good pebble guidance performance remain to be done.

Booster tracking algorithms can:

- Estimate so-called tactical parameters, which are the missile launch locations, times, azimuths, altitudes, and the degree to which the missile is lofted or depressed from a nominal trajectory;
- Assess the number and types of missiles launched in the raid;
- Predict missile payload impact points on the earth;
- cue midcourse and terminal ballistic missile defense systems, both sensors and interceptors; and
- Provide fire control information for booster interception.

Tactical parameter estimation, raid assessment, and coarse impact point prediction constitute the traditional tactical warning and attack assessment (TW/AA) functions. More precise impact point predictions and cues to midcourse and terminal sensors and interceptors can enhance midcourse and terminal BMD performance and also enhance TW/AA performance.

There are essentially three data processing or filtering methods for tracking boosters. The first method is to model in detail the complex three-dimensional dynamics of the booster. This requires some knowledge of key booster parameters such as thrust, mass, and drag. The full art of Kalman filtering technology is required to compensate for model errors such as uncertain and neglected parameters in the booster dynamics models and booster maneuvers. The Kalman filter's integration of the booster's three-dimensional equations of motion and estimation of the booster's three-dimensional trajectory using the sensor data, if it is achievable, reliable, and credible, should provide the best tracking performance.

For the Kalman filter booster tracking algorithm, the dynamics model is critical. Booster dynamics are complex, involving gravity, time varying propulsive forces, and aerodynamic drag and lift forces, which vary with the missile type and mod and with the booster's velocity and angle of attack. Detailed models used in trajectory construction and reconstruction are not appropriate to tracking because they depend on too many individual parameters, which cannot be estimated accurately, reliably, and credibly in real time. It should not be necessary, however, to model and estimate these many parameters to accurately track the target for the purposes of BMD. Neglecting some parameters, or modeling their effect approximatel;' or incorrectly will introduce model errors in the Kalman filter. These must be compensated by advanced Kalman filtering techniques.

Missile booster trajectories are principally determined by propulsive capabilities and guidance sophistication. In general, missile boosters both as a matter of course and as a matter of design can execute very significant maneuvers, including changes between orbital planes (yaw), staging and other thrust variations, and maneuvers designed to burn off excess propulsive energy (so-called energy management maneuvers) such as nonzero angle-of-attack flight profiles and in-plane (pitch) changes. These maneuvers can also be considered as model errors in the Kalman filter that also are to be compensated for by advanced Kalman filtering techniques.

High fidelity, data-adaptive tracking filters are by their nature designed to chase boosters through significant maneuvers and model errors. Yet it remains to be demonstrated that boost phase tracking filters using angles-only measurements can track boosters accurately through pitch and yaw changes, staging and other thrust variations, energy management maneuvers, and modeling errors, in a possibly dense observation environment.

The second method for booster tracking is to determine tl. ee-dimensional booster positions by associating and triangulating lines-of-sight across multiple satellites. Three-dimensional velocity is computed by interpolation and numerical differentiation of the positions. Compared to integrating the three-dimensional equations of motion, the interpolation and numerical differentiation are inherently limited in the accuracy of their velocity estimates, although it remains to be seen whether this accuracy is sufficient for BMD.

The last method is to use a nonlinear iterative least squares algorithm to fit the angles-only sensor data to *a priori* booster altitude versus ground range flight profiles, where it is assumed that the booster's motion with respect to the launch point is nearly fixed in one plane. Template matching depends on the validity of the *a priori* assumptions on altitude versus ground range flight profiles and the essentially two-dimensional character of the trajectories. Unsophisticated guidance methods would tend to keep booster trajectories to historical average flight profiles and essentially two-dimensional.

To initialize the Kalman filter the nonlinear iterative least squares algorithm fits the angles-only sensor measurement data from an early portion of the powered-flight (60-120 seconds time after launch is common) to *a priori* booster altitude versus ground range flight profiles. The least squares fit estimates the booster's three-dimensional state at the time of the last measurement, the current time say, which initializes the Kalman filter, for it to maintain track beyond this initial period.

The same least squares fit estimates the booster's tactical parameters. These are the booster's initial state, which consists of launch latitude and longitude, launch time, launch azimuth, launch altitude, and the degree to which the booster is lofted or depressed from some nominal trajectory. An estimated launch region is determined from the estimation error. Estimation of tactical parameters is an important part of tactical warning and attack assessment.

TW/AA also includes decisions on missile type. Missile typing is important for BMD battle management and may be important in booster tracking algorithms by helping to determine values for booster thrust, mass, and drag used in Kalman tracking filters. A template matching is performed in which sensor intensity measurement data is fit to intensity versus time profiles. Missile type is decided according to the intensity template that achieves the best fit.

The angles-only data template matching also performs missile typing. One method is to test which missile types can be found in the estimated launch region by comparison to a data base of missile launcher locations or mobile missile patrol areas. Another method is to select the missile type according to the template that gives the best fit of the angles-only data to the altitude versus ground range profiles. The latter is referred to as metric typing.

In the execution of template matching algorithms these specific questions remain to be quantified:

- Under what viewing geometry and range conditions does the numerical algorithm converge?
- What is the accuracy and precision of the initial and current state estimate?

- What are the effects of clutter, false alarms, and multiple targets on convergence and estimation performance?
- How accurately can missile typing decisions be made and what is the relative importance of the different missile-typing performance algorithms?

6.2.1 To Which Template Should the Observed Data Be Matched?

There are essentially four sources for templates. First, the Aerospace Corporation publishes the Master Target Model Book which is a summary of booster flight characteristics based on data observed by national technical means and trajectory reconstruction programs. Typically, there is one and only one set of profiles for each missile type and mod. No lofting or depression information is present.

The missile systems in the Model book have already flown in contrast to the future systems that typically populate SDIO threat scenarios, which are set in the future. For these future missiles, in an apparently ad hoc manner, the SDIO threat generation community has produced templates using the same software that produces booster trajectories for the threat scenarios. Because they are made to order, in a sense, a family of templates for each missile type and mod can be developed to represent lofting and depression from nominal trajectories.

A third source is when some contractors with sophisticated booster modeling capabilities augment whatever templates they are given from whatever source to create lofted and depressed templates. This is fine so long as some check is maintained to prevent the templates from being tuned to match the trajectories they are to track.

The final source is when threat trajectories in a particular attack scenario on a particular threat tape are averaged into templates. Clearly use of this sort of template is suspect. The immediate relationship of the booster trajectories in the threat with the templates that are then used to track them, the very same trajectories that generated the templates, produces possibly invalid results and grossly misleading performance assessments. This source of templates must be strongly discouraged.

The Master Target Model Book cannot supply the templates for SDIO scenarios set in the future because of the mismatch between templates for missile systems that have been observed, on the one hand, and SDIO threat scenarios that involve future missile systems that have yet to fly and for which no templates based on observed data can be constructed, on the other. The community that produces the Model Book could be asked to produce templates for future systems but this would represent a departure from their standard methods, most importantly the reliance on observed data. SDIO, possibly by way of its Threat Working Group, possibly in conjunction with the intelligence community, needs to firmly control the development and promulgation of templates for use by SDIO contractors. There currently is a gap of immense proportions between the intelligence community providing templates on current inventory missiles and the free-for-all of assumptions on the character and content of template data being made by SDIO contractors. If control is not taken by SDIO, the validity of all template matching results is at risk and could be considered suspect.

One key issue is the diversity of templates and the uncertainty of their data. Templates are *a priori* assumptions for missile flight trajectories. Templates are also averages over observed, historical flight profiles. There are, however, no uncertainty bars around their data. How uncertain template data are because of uncertainties in the observed data and trajectory reconstruction processing, how much templates vary by the source that produces them and by missile type and missile mod and whether it matters are all open questions. The usual assumption made within the SDIO community is to use a bank of templates to account for any wide diversity, with the tracking algorithm selecting the correct template for a particular target and the particular template within a family of lofted to depressed altitude versus ground range flight profiles for the missile type and mod.

The most important issue is the degree to which templates are identical to the trajectories being tracked. Should they match? If the altitude versus ground range templates closely parallel the booster trajectories to be tracked then good template matching performance should be expected. In a sense, close identity is akin to having assumed away the problem: All the uncertainty in the booster motion is removed and captured in the *a priori* data bases of templates. Unless we believe we can assume away the problem, templates, in general, should not be identical to the actual booster trajectories to be tracked. One exception might be third world missile forces whose unsophisticated guidance systems would keep the booster trajectories to simple flight profiles.

6.2.2 Templates and the Use of *a priori* Information

For some, use of *a priori* information should be very limited or even avoided entirely. In response, it can be argued that the modeling assumptions that enter into tracking filters are themselves a form of *a priori* information. These assumptions may include supposing that the booster jerk (the rate of change of the acceleration) is constant, or that the booster will fly a gravity turn in first stage, or that the thrust is constant for each stage and that its magnitude is known, or that the booster will fly along *a priori* flight profiles. But there are differences among types of *a priori* information both in content and effect.

A priori information can serve usefully to restrict target dynamics models and compensate for information not present in measurements. For instance, in the model for the deterministic and stochastic components of the dynamics, the algorithm designer relies on his computational experience and intuition to govern what must be included and how it is included, and what can be neglected. Deciding whether and how to model variations in thrust, for instance, is different in scale from assuming the booster will fly in the manner of *a priori* flight profiles.

The key concern is that reliance on *a priori* information may leave the algorithm vulnerable to boosters that do not do the expected, or the mean, or are of type and circumstance outside the *a priori* information's domain of applicability. The consequence may be susceptibility to catastrophic failure. Balance is the key: use *a priori* information when necessary but in a manner that is as flexible as possible and that does not leave undue susceptibility to catastrophic failure.

6.2.3 The Concept of Tracking Algorithm Architecture

Tracking algorithm architecture refers to the structure and flow of information within a tracking algorithm. The SDI Tracking Panels have identified four basic tracking algorithm architectures:

Type I: Individual sensors operate independently.

Type II: Individual sensors develop tracks independently that are then fused across pairs and multiples of sensors and possibly fed back to the individual sensors.

Type III: Observations are fused across pairs and multiples of sensors and then processed into tracks.

Type IV: Observations and tracks are processed centrally after association of observations and tracks is performed at each sensor.

Each architecture has its particular strengths and weaknesses. Type I architectures are the most survivable, have the least communication loads and simplest operational needs, but must track within the limitations of the information provided by only one sensor. For instance, passive electrooptical sensors measure the angles-only information in the line of sight to a target and do not measure range. Angles-only measurements provide scant information by which to estimate booster dynamics, which are quite variable. This single satellite angles-only booster tracking filtering problem is a challenge to the state of the art.

Range can be obtained by fusing line-of-sight measurements across pairs of satellites as in Type III algorithms. Fusion of lines of sight across pairs of satellites demands large communication loads, and algorithms that rely entirely on having fused lines-of-sight information are the least survivable. Moreover, the association of lines of sight across stereo partners is a challenge to the state of the art in dense observation environments arising from multiple targets, clutter, and false alarms.

Type II algorithms can be subdivided into two groups based on whether the fused multiple sensor tracks are fed back to the individual sensors. It is also useful to distinguish whether the tracks being fused are focal plane tracks or full state tracks. Focal plane track fusion has the same association challenges as fusion of lines-of-sight. Fused full state tracks fed back to the sensor to replace each satellite's focal plane tracks tracks would then be maintained by the satellites. Full state track fusion is less challenging owing to the additional information in the full state which assists greatly in the association process. Fusion of multiple satellite full state tracks is done to enhance the quality of the individual sensor full state estimates and to provide a comprehensive view of the surveillance region.

Type IV algorithms are ideal in terms of performance. Each sensor associates observations and tracks, which are then passed on for centralized data processing. Tracks from the centralized data processor are used by the individual sensors for the association. The estimation performance is ideal because all available data are used in the processing. Communication demands are very high. Operational demands typically are so great as to make these algorithms impractical, however.

In BMD tracking algorithm development work to date, Type I and Type II algorithms dominate, the latter mostly without feedback. There has been some development work on Type III algorithms. We are not familiar with any credible Type IV approaches.

6.3 SINGLE SENSOR MIDCOURSE ANGLES-ONLY TRACK INITIATION

In midcourse track initiation, the targets are expected to be closely spaced in the real three-dimensional space and, more importantly, on sensor focal planes, depending on sensor resolution and signal processing and on sensor-object distance and aspect angle. Closely spaced objects are expected to generate a large number of candidate track assemblies, many with poor track purity. Impure track assemblies may suffer numerical divergence or inhibited numerical convergence and inaccurate convergence values. The immense cost in midcourse cold-start track initiation arises from the assembly of a large number of candidate tracks, the computational expense of computing initial state estimates for each, and the computational and memory burdens for storing and sorting candidate tracks.

Computing an initial state estimate from single sensor angles-only data of an object in orbit has a long and distinguished pedigree in celestial mechanics, where it is referred to as the problem of initial orbit determination. We shall describe the classical methods of initial orbit determination of Laplace and of Gauss, including the Herrick-Gibbs refinement to Gauss's method, that emphasize using the minimum possible data set of three sets of azimuth-elevation angle measurements. We shall also describe the estimation-based angles-only initial orbit determination methods of Chang that use more than three measurement sets. Last, we shall describe a new algorithm proposed by Taff et al. that is a new approach to initial orbit determination.

6.3.1 Classical Angles-Only Initial Orbit Determination: Laplace and Gauss and Herrick-Gibbs

The classical methods of initial orbit determination from three sets of angles-only measurements were published by Laplace in 1780[1] and Gauss in 1809.[2] Both methods lead to equations of similar form:

$$\rho = A + \frac{B}{r^3}$$

$$r^2 = \rho^2 + R^2 + 2\,c\,\rho\,R \quad , \qquad (6\text{-}1)$$

1 According to Laurence G. Taff in *Celestial Mechanics.*

2 Gauss, K.F., *Theoria Motus Corporum Coelestium*, 1809 reprinted as *Theory of Motion of the Heavenly Bodies*, New York, Dover, 1963.

which leads to the eighth-order equation for r:

$$r^8 = (A^2 + R^2 + 2\,c\,A\,R)\,r^6 + 2B\,(A + c\,R)\,r^3 + B^2 \quad . \tag{6-2}$$

In these equations, ρ is the unknown sensor-object distance, r is the unknown object-earth distance, R is the known sensor-earth distance, c is a known number, and A and B are coefficients determined by, and particular to, Laplace's method and Gauss's method.

Laplace's Method

Laplace's method uses the exact dynamics of a satellite orbiting a spherical earth, the line-of-sight vectors determined by the angular measurements, and numerical estimates of the first and second time derivatives of the line-of-sight vectors. Consider the object located at $\underline{r}$ with respect to an earth-centered inertial (ECI) coordinate frame, the observing sensor located at $\underline{R}$ in the same ECI coordinates, and the object located at $\underline{\rho}$ with respect to a satellite-centered nonrotating frame.[3] Thus,

$$\underline{r} = \underline{\rho} + \underline{R} \quad . \tag{6-3}$$

For a spherical earth, Newton's law of gravity states that

$$\ddot{\underline{r}} = -\frac{\mu\,\underline{r}}{r^3} \ , \ \ddot{\underline{R}} = -\frac{\mu\,\underline{R}}{R^3} \quad . \tag{6-4}$$

Now $\underline{\rho} = \rho\underline{L}$, where $\underline{L}$ is the line-of-sight unit vector determined by the azimuth and elevation angles. With Newton's law Eq. (6-5) becomes

$$\begin{aligned} \ddot{\underline{r}} &= -\frac{\mu\,(\rho\underline{L} + \underline{R})}{r^3} = \ddot{\underline{\rho}} + \ddot{\underline{R}} \\ &= \ddot{\rho}\,\underline{L} + 2\,\dot{\rho}\,\dot{\underline{L}} + \rho\,\ddot{\underline{L}} - \frac{\mu\,\underline{R}}{R^3} \quad . \end{aligned} \tag{6-5}$$

3 Vector quantities are denoted by underline.

By the following operation

$$(\underline{L} \times \dot{\underline{L}}) \cdot \frac{-\mu(\rho\,\underline{L} + \underline{R})}{r^3} = \rho\,(\underline{L} \times \dot{\underline{L}}) \cdot \ddot{\underline{L}} + (\underline{L} \times \dot{\underline{L}}) \cdot \frac{-\mu\,\underline{R}}{R^3}$$

$$(\underline{L} \times \ddot{\underline{L}}) \cdot \frac{-\mu(\rho\,\underline{L} + \underline{R})}{r^3} = 2\,\dot{\rho}\,(\underline{L} \times \ddot{\underline{L}}) \cdot \dot{\underline{L}} + (\underline{L} \times \ddot{\underline{L}}) \cdot \frac{-\mu\,\underline{R}}{R^3} \qquad (6\text{-}6)$$

we can isolate ρ and dρ/dt:

$$\rho = -\frac{1}{(\underline{L} \times \dot{\underline{L}}) \cdot \ddot{\underline{L}}}\left[\frac{\mu}{r^3}(\underline{L} \times \dot{\underline{L}}) \cdot \underline{R} - \frac{\mu}{R^3}(\underline{L} \times \dot{\underline{L}}) \cdot \underline{R}\right]$$

$$\dot{\rho} = -\frac{1}{2(\underline{L} \times \ddot{\underline{L}}) \cdot \dot{\underline{L}}}\left[\frac{\mu}{r^3}(\underline{L} \times \ddot{\underline{L}}) \cdot \underline{R} - \frac{\mu}{R^3}(\underline{L} \times \ddot{\underline{L}}) \cdot \underline{R}\right] \quad . \qquad (6\text{-}7)$$

There are two critical issues in using Laplace's method. First, the accuracy with which the time derivatives of $\underline{L}$ can be numerically computed from measurements of three lines of sight. Second, the method fails when

$$(\underline{L} \times \dot{\underline{L}}) \cdot \ddot{\underline{L}} = 0 \quad .$$

This is necessarily the case when the sensor satellite and target are coplanar.

Gauss's Method

From classical mechanics we know the motion of objects under the influence of a central force (a force acting always along the line connecting the object to the force center) is always motion in a plane: the angular momentum vector $\underline{h} = \underline{r} \times \underline{v}$ is constant. Gauss's method invokes the coplanar character of the three positions in the angles-only observations and uses an analytical approximation for a power series solution of the equations of motion.

Consider the three three-dimensional positions $\underline{r}_1$, $\underline{r}_2$, and $\underline{r}_3$. Because the objects are coplanar, the three position vectors are necessarily linearly dependent, that is, there exist scalars c_1 and c_3 such that

$$\underline{r_2} = c_1\,\underline{r_1} + c_3\,\underline{r_3} \quad . \qquad (6\text{-}9)$$

Now, we can compute

$$
\begin{aligned}
\underline{r_1} \times \underline{r_2} &\equiv 2\, A_{12}\underline{w} \\
&= c_3\, \underline{r_1} \times \underline{r_3} \equiv 2\, c_3\, A_{13}\underline{w} \\
\underline{r_3} \times \underline{r_2} &\equiv -2\, A_{23}\underline{w} \\
&= -c_1\, \underline{r_1} \times \underline{r_3} = -2\, c_1\, A_{13}\underline{w}
\end{aligned}
\tag{6-10}
$$

where the A_{ij}'s are the areas of the triangles formed by the vectors and $\underline{w}$ is a unit vector in the direction of the angular momentum vector: $\underline{w} = \underline{h} / |\underline{h}|$. We immediately find that

$$c_3 = A_{12}/A_{13}\ ,\ c_1 = A_{23}/A_{13}\ . \tag{6-11}$$

Below we shall derive the power series solution of the equations of motion, which take the form

$$\underline{r_1} = F_1\, \underline{r_2} + G_1\, \underline{v_2}\ ;\ \underline{r_3} = F_3\, \underline{r_2} + G_3\, \underline{v_2} \tag{6-12}$$

where the F and G coefficients are often referred to as Lagrange's F and G coefficients. Using this result we find

$$
\begin{aligned}
A_{12} &\equiv \frac{1}{2}\, \underline{w} \cdot (\underline{r_1} \times \underline{r_2}) = -\frac{1}{2}\, G_1\, \underline{w} \cdot (\underline{r_2} \times \underline{v_2}) = -\frac{1}{2}\, G_1 h \\
A_{23} &\equiv \frac{1}{2}\, \underline{w} \cdot (\underline{r_2} \times \underline{r_3}) = \frac{1}{2}\, G_3 h \\
A_{13} &\equiv \frac{1}{2}\, \underline{w} \cdot (\underline{r_1} \times \underline{r_3}) = \frac{1}{2}\, (F_1\, G_3 - F_3\, G_1) h
\end{aligned}
\tag{6-13}
$$

since $\underline{h} = \underline{r}_2 \times \underline{v}_2 = |\underline{r}_2 \times \underline{v}_2|\, \underline{w}$. Therefore,

$$c_3 = -\frac{G_1}{F_1\, G_3 - F_3\, G_1}\ ,\ c_1 = -\frac{G_3}{F_1\, G_3 - F_3\, G_1}\ . \tag{6-14}$$

Now the power series solution of the equations of motion

$$\ddot{\underline{r}} + \frac{\mu}{r^3}\,\underline{r} = 0 \tag{6-15}$$

is based on a Taylor series expansion for r about some $\underline{r}_0$

$$\underline{r}(t) = \sum_{n=0}^{\infty} \frac{(t-t_0)^n}{n!} \; \frac{d^{(n)}\underline{r}}{dt^n}\Bigg|_{(t=t_0)} \tag{6-16}$$

Successive differentiation of the equations of motion involve higher order derivatives of μ/r^3:

$$\begin{aligned}
\epsilon \triangleq \frac{\mu}{r^3} \;&\Rightarrow\; \frac{d\epsilon}{dt} = -\frac{3\mu}{r^4}\frac{dr}{dt} = -\frac{3\epsilon}{r}\frac{dr}{dt} = -3\epsilon\lambda \\
\lambda \triangleq \frac{1}{r}\frac{dr}{dt} &= \frac{\underline{r}\cdot\underline{v}}{r^2} \\
&\Rightarrow \frac{d\lambda}{dt} = \frac{v^2}{r^2} + \frac{\underline{r}}{r^2}\cdot(-\epsilon\,\underline{r}) - \frac{2(\underline{r}\cdot\underline{v})}{r^3}\frac{dr}{dt} \\
&\qquad = \psi - \epsilon - 2\lambda^2 \\
\psi \triangleq \frac{v^2}{r^2}& \\
&\Rightarrow \frac{d\psi}{dt} = 2\frac{\underline{v}}{r^2}\cdot\frac{d\underline{v}}{dt} - 2\frac{v^2}{r^3}\frac{dr}{dt} \\
&\qquad = -2\epsilon\lambda - 2\psi\lambda \;.
\end{aligned} \tag{6-17}$$

We see that ϵ, λ, and ψ form a closed set under differentiation. We can now evaluate the derivatives in the series expansion in terms of these expressions

$$\begin{aligned}
&\frac{d\underline{r}}{dt} = \underline{v} \;, \quad \frac{d^2\underline{r}}{dt^2} = \frac{d\underline{v}}{dt} = -\epsilon\,\underline{r} \\
&\frac{d^{(3)}\underline{r}}{dt^3} = -\dot{\epsilon}\,\underline{r} - \epsilon\,\dot{\underline{r}} = [3\epsilon\lambda]\underline{r} - [\epsilon]\underline{v} \\
&\frac{d^{(4)}\underline{r}}{dt^4} = 3\dot{\epsilon}\lambda\underline{r} + 3\epsilon\dot{\lambda}\underline{r} + 3\epsilon\lambda\underline{v} - \dot{\epsilon}\underline{v} - \epsilon\dot{\underline{v}} \\
&\qquad = [-15\epsilon\lambda^2 - 2\epsilon^2 + 3\epsilon\psi]\underline{r} + [6\epsilon\lambda]\underline{v} \\
&\text{etc.}
\end{aligned} \tag{6-18}$$

In this fashion, the Taylor series expansion becomes

$$\underline{r}(t) = F(t)\underline{r}_0 + G(t)\underline{v}_0 \tag{6-19}$$

where $\underline{r}_0$ and $\underline{v}_0$ are constants.

In Gauss's method the Taylor series expansion is about the intermediate state $\underline{r}_2$ and $\underline{v}_2$ and the analytical approximation truncates the Lagrange coefficients after the first two terms

$$\begin{aligned} F_i &= 1 - \frac{\epsilon_2}{2}(t_i - t_2)^2 + \ldots \\ G_i &= (t_i - t_2) - \frac{\epsilon_2}{3!}(t_i - t_2)^3 + \ldots \ . \end{aligned} \tag{6-20}$$

The coefficients c_1 and c_2 are now easily expressed in terms of the times and $\underline{r}_2$. Substituting Eq. (6-3) and the expressions for the c's into the coplanarity condition Eq. (6-9), we can derive Eq. (6-1) for r_1 and ρ_2. We can then determine $\underline{r}_2$ after which we can compute $\underline{r}_1$ and $\underline{r}_3$.

To complete the initial orbit determination problem, we need an expression for $\underline{v}_2$. One method is to interpolate among $\underline{r}_1$, $\underline{r}_2$, and $\underline{r}_3$ and then numerically differentiate. The Herrick-Gibbs[4] improvement to this method is to truncate the power series after the fourth order term. With the position vectors as determined above we can compute an expression for $\underline{v}_2$ that is valid to fourth-order in time.

Gauss's method performs poorly when the analytic approximations for the power series solutions to the equations of motion are poor, and hence the values for the c coefficients are poor. Whether this prevents the application of this method for ballistic missiles is currently a matter of contention.

[4] Gibbs, W.J., Mem. Nat'l Acad. Sci., 4 (1888) and Herrick, S., *The Laplacian and Gaussian Orbit Methods*, University of California Press, Vol. 1, No. 1, 1940.

6.3.2 Estimation-Based Initial Orbit Determination: Chang

Chang[5] developed an iterative least square algorithm for estimating the state of a nonlinear deterministic system with nonlinear noisy measurements which he applied to the problem of angles-only initial orbit determination using more than three observations. Following Chang consider the nonlinear discrete system:

$$\underline{x}_{n+1} = \underline{f}(\underline{x}_n), \quad n=1,... \tag{6-21}$$

or nonlinear continuous system

$$\dot{\underline{x}} = g(\underline{x}) \tag{6-22}$$

and the nonlinear measurement equations:

$$\underline{y}_{n+1} = \underline{h}(\underline{x}_{n+1}) + \underline{v}_{n+1} \tag{6-23}$$

where $\underline{x}$ is the state vector, $\underline{y}$ is the noise corrupted measurement vector, $\underline{v}$ is the white Gaussian zero mean measurement noise process with covariance R_n, and n is the discrete time index. We can always obtain an equivalent discrete system from the continuous system by numerical integration of the equation of motion:

$$\begin{aligned} \underline{x}(t_{n+1}) &= \underline{x}(t_n) + \int_{t_n}^{t_{n+1}} g(\underline{x})\,dt \\ &= \underline{f}(\underline{x}_n) \end{aligned} \tag{6-24}$$

We can relate the state vector at time n to the state vector at time 1 by iterating the equation of motion n times

$$\underline{x}_n = \underline{f}_n(\underline{x}_1) \tag{6-25}$$

---

5 C.B. Chang, *Optimal State Estimation of Ballistic Trajectories with Angle-Only Measurements*, MIT Lincoln Laboratory Technical Note 1979-1, 24 January 1979 and "Ballistic Trajectory Estimation with Angle-Only Measurements," *IEEE Trans. Auto. Contr.*, Vol. AC-25, No. 3, June 1980, pp. 474-480.

where if the system were linear $f_n(\ )$ would be the product of n transition matrices and if the system were continuous then

$$\underline{f}_n(\underline{x}_1) = \underline{x}(t_1) + \int_{t_1}^{t_n} \underline{g}(\underline{x})\,dt \quad . \tag{6-26}$$

The algorithm iteratively processes a batch of data, $\underline{y}_1, \underline{y}_2, \ldots, \underline{y}_n$, to determine an estimated state sequence for $\underline{x}_n$, n = 1, ..., N that, subject to the constraints of the equations of motion, minimizes the weighted least squares error:

$$J = \sum_{n=1}^{N} (\underline{y}_n - \underline{h}(\underline{x}_n))^T R_n^{-1} (\underline{y}_n - \underline{h}(\underline{x}_n)) \tag{6-27}$$

The minimization is accomplished by expanding the nonlinear system and measurement equations in a first order Taylor series about an initial estimate of the true state:

$$\begin{aligned}
\underline{x}_n &= \underline{f}_n(\underline{x}_1) \approx \hat{\underline{x}}_n^0 + F_n^0(\underline{x}_1 - \hat{\underline{x}}_1^0) \\
\hat{\underline{x}}_n^0 &= \underline{f}_n(\hat{\underline{x}}_1^0) = \text{initial estimate for } \underline{x}_n \\
\hat{\underline{x}}_1^0 &= \text{initial estimate for } \underline{x}_1 \\
F_n^0 &= \left.\frac{\partial \underline{f}_n(\underline{x}_1)}{\partial \underline{x}_1}\right|_{\underline{x}_1 = \underline{x}_1^0} = \text{Jacobian of } \underline{f}_n(\underline{x}_1) \text{ evaluated at } \underline{x}_1^0
\end{aligned} \tag{6-28}$$

$$\begin{aligned}
\underline{h}(\underline{x}_n) &\approx \underline{h}(\hat{\underline{x}}_n^0) + H_n^0(\underline{x}_n - \hat{\underline{x}}_n^0) \\
&= \underline{h}(\hat{\underline{x}}_n^0) + H_n^0 F_n^0(\underline{x}_1 - \hat{\underline{x}}_1^0) \\
H_n^0 &= \left.\frac{\partial \underline{h}(\underline{x}_n)}{\partial \underline{x}_n}\right|_{\underline{x}_n = \hat{\underline{x}}_n^0} = \text{Jacobian of } \underline{h}(\underline{x}_n) \text{ evaluated at } \hat{\underline{x}}_n^0 \ .
\end{aligned} \tag{6-29}$$

The minimization equation now reads

$$\begin{aligned}
J \approx \sum_{n=1}^{N} &\left(\underline{y}_n - [\underline{h}(\hat{\underline{x}}_n^0) + H_n^0 F_n^0(\underline{x}_1 - \hat{\underline{x}}_1^0)]\right)^T R_n^{-1} \\
&\cdot \left(\underline{y}_n - [\underline{h}(\hat{\underline{x}}_n^0) + H_n^0 F_n^0(\underline{x}_1 - \hat{\underline{x}}_1^0)]\right) \ .
\end{aligned} \tag{6-30}$$

Taking the derivative of J with respect to $\underline{x}_1$ and solving for $\underline{x}_1$ yields

$$\left.\frac{\delta J}{\delta \underline{x}_1}\right|_{\underline{x}_1=\hat{\underline{x}}_1^1} = -2\sum_{n=1}^{N} F_n^{0T} H_n^{0T} R_n^{-1}\left[\underline{y}_n - \underline{h}(\hat{\underline{x}}_n^0) - H_n^0 F_n^0(\hat{\underline{x}}_1^1 - \hat{\underline{x}}_1^0)\right] = 0$$

$$\Rightarrow \hat{\underline{x}}_1^1 = \hat{\underline{x}}_1^0 + \left[\sum_{n=1}^{N} F_n^{0T} H_n^{0T} R_n^{-1} H_n^0 F_n^0\right]^{-1} \left[\sum_{n=1}^{N} F_n^{0T} H_n^{0T} R_n^{-1}(\underline{y}_n - \underline{h}(\hat{\underline{x}}_n^0))\right] \quad . \tag{6-31}$$

If the algorithm converges, the solution is a better estimate than the initial estimate. If we use the solution in the place of the initial estimate and repeat the calculation, and if the algorithm converges, we will compute a further improved estimate. We have thus derived an iterative algorithm:

$$\hat{\underline{x}}_1^{k+1} = \hat{\underline{x}}_1^k + \left[\sum_{n=1}^{N} F_n^{kT} H_n^{kT} R_n^{-1} H_n^k F_n^k\right]^{-1} \left[\sum_{n=1}^{N} F_n^{kT} H_n^{kT} R_n^{-1}(\underline{y}_n - \underline{h}(\hat{\underline{x}}_n^k))\right] \tag{6-32}$$

and the covariance of the estimate is

$$\mathrm{cov}\,(\hat{\underline{x}}_1^{k+1}) = \left[\sum_{n=1}^{N} F_n^{kT} H_n^{kT} R_n^{-1} H_n^k F_n^k\right]^{-1} \quad . \tag{6-33}$$

This follows from rewriting the estimate as

$$\begin{aligned}\hat{\underline{x}}_1^{k+1} &= \hat{\underline{x}}_1^k + \left[\sum_{n=1}^{N} F_n^{kT} H_n^{kT} R_n^{-1} H_n^k F_n^k\right]^{-1} \\ &\qquad \cdot \left[\sum_{n=1}^{N} F_n^{kT} H_n^{kT} R_n^{-1}(\underline{v}_n + H_n^k F_n^k(\underline{x}_1 - \hat{\underline{x}}_1^k))\right] \\ &= \underline{x}_1 + \left[\sum_{n=1}^{N} F_n^{kT} H_n^{kT} R_n^{-1} H_n^K F_n^k\right]^{-1} \left[\sum_{n=1}^{N} F_n^{kT} H_n^{kT} R_n^{-1} \underline{v}_n\right]\end{aligned} \tag{6-34}$$

The process is terminated when the difference in values of J between successive iterations is below a threshold. From the estimate and covariance at the initial time, the estimates at any time can be calculated as

$$
\begin{aligned}
\hat{\underline{x}}_n &= \underline{f}_n(\hat{\underline{x}}_1) \\
&\approx \underline{f}_n(\underline{x}_1) + \left.\frac{\partial \underline{f}_n}{\partial \underline{x}_1}\right|_{\underline{x}_1 = \hat{\underline{x}}_1} (\underline{x}_1 - \hat{\underline{x}}_1) \\
\Rightarrow \text{cov}\,(\hat{\underline{x}}_n) &= F_n \,\text{cov}\,(\hat{\underline{x}}_1)\, F_n^T
\end{aligned}
\qquad (6\text{-}35)
$$

The convergence of the algorithm depends strongly on the initial guess for the iterations. Chang suggested two procedures for computing the initial guess. One method is to smooth the batch of angle measurements by a second order polynomial to obtain their derivatives. The smoothed angles and derivatives are applied to the equations of motion to compute the range and range rate, which together form the initial guess. This algorithm is a modification of Laplace's method that uses more than three observations. Its computation of range and range rate fail when the sensor and target are coplanar, as does Laplace's method.

The second method uses the fact that total mechanical energy is conserved during the free-fall portion of flight. An alternative equation for range and range rate can be derived from the energy equation. We can compute range and range rate with a guess for the average total energy and using the smoothed angles and derivatives. Thrasher[6] discusses a method for detecting the inplane case.

Herget's method, which also uses more than three angles-only observations in a least squares estimation algorithm, predates Chang's algorithm.[7]

6.3.3 Statistical Initial Orbit Determination: Taff et al.

Taff et al.[8] adopt a very different point of view compared to the methods presented above. Indeed, they regard the computation of initial orbits based on angles-only data as a "futile endeavor." Their argument is essentially that the high observation density makes track assembly and cold start track initiation algorithms too expensive to implement. And furthermore, the initial state estimates they generate are of poor enough precision that

6 Roy Thrasher, *State Estimation of Ballistic Trajectories with Angle-Only Measurements*, Alien Research Corporation Report ARC-TR-87-003, 21 May 1987.

7 Paul Herget, "Computation of Preliminary Orbits," *The Astronomical Journal*, Vol. 70, pp. 1-3, 1965.

8 L.G. Taff, B. Belkin, and G.A. Schweter, "Statistical Initial Orbit Determination," *Proceedings of the SDI Panels on Tracking*, Issue No. 3, 1990, pp. 4-285 to 4-307.

subsequent data association attempts are complex and expensive, and ultimately do not give good performance. Taff et al.'s method uses templates to represent all possible orbits from a multitude of launch points to a multitude of impact points. The angles-only observational data would then be used to select among the templates which have good estimation precision.

Taff et al. begin by dividing the Soviet Union into rectangles and similarly dividing the conterminous United States, Alaska, and Hawaii. They develop a set of orbits representing flights from each launch area into each impact area. The algorithm first selects from among all possible orbits the set that best matches the first angles-only observation. Then with the next angles-only observation, the algorithm selects the best match from the subset. This results in one particular orbit.

This method is very interesting. By computing the initial orbits with essentially two angles-only observations it avoids the great challenge of cold-start track assembly. Initial orbits could be determined for any two observations from successive frames. The algorithm's feasibility remains to be demonstrated, however.

7. PERFORMANCE EVALUATION METHODS FOR MULTIPLE TARGET TRACKING ALGORITHMS[1]

Ultimately, the performance of tracking algorithms is judged by the success, or failure, of the mission they support. The destruction of a target by an interceptor guided, in part, by tracking information provides one vivid, obvious measure of success. But, what if the interceptor missed? Did the tracking algorithm perform poorly, or the guidance algorithm, or the sensor and signal processing, or the rocket motor?

In a computer simulation of a complex system comprised of myriad subsystems and algorithms, it is difficult without specific tests to untangle the performance of any individual component. We describe measures of effectiveness to evaluate tracking algorithm performance in computer simulations. While these might be considered to be intermediate measures of effectiveness for the system as a whole, they are all important for diagnosing and evaluating tracking algorithms considered in their own right.

Evaluation of tracking performance is straightforward in an environment of few, widely spaced targets and no false alarms or clutter. In this sparse environment, a track is consistently updated with measurements from the same target. The track, or state estimate, is then associated and compared with the true state of the target, which is obvious as identified by the one source of the measurements.

Performance evaluation is more complex in a dense environment of:

- False alarms;
- Clutter;
- Multiple targets;
- Individual observations arising from unresolved closely spaced objects (CSOs).

In this case, a track is not consistently updated with measurements from the same target because some sensor observations of other targets, clutter, or false alarms will be

---

[1] This chapter is a close adaptation of a paper co-authored with Oliver E. Drummond, Air Defense Systems Division, General Dynamics, and presented at the SPIE Conference Signal and Data Processing of Small Targets 1991, 1-3 April 1991, Orlando, Florida.

incorrectly associated with the track and some sensor observations associated with the track will be of unresolved CSOs. With misassociations and unresolved CSOs, the source of the measurements in a track will not be a clear indication of a single target, thus confusing which track is to be compared with the true state of a target.[2,3,4,5] Furthermore, in a dense target environment, there may be

- Missed tracks: targets without tracks;
- Redundant tracks: more than one track for one target;
- Spurious tracks: tracks for no targets whatsoever.

We describe scoring methods for evaluating the performance of multiple target tracking (MTT) algorithms fairly without undue bias towards any particular type. For instance, some algorithms may generate many "extra" tracks, such as in multiple hypothesis tracking, but the track purity and state estimation accuracy of the N best are better than the N tracks of algorithms that do not knowingly generate "extra" tracks, such as local nearest neighbor. Insofar as track purity and state estimation are concerned, the former is to be preferred, whereas the latter may be preferred from the standpoint of computational and memory costs and size, weight, and power of on-board processors.

These methods were developed initially by individuals and further developed and adapted by the members of the SDI Panels on Tracking. It is part of an ongoing process and is not to be considered as the last word on the subject.

Track purity over a time interval refers to the degree to which a track's measurements over that time originate from a particular target. In single target tracking without false alarms and clutter, track purity is ensured and the association of track-to-truth unambiguous. Multiple target tracking typically involves many impure tracks and, therefore, ambiguous track-to-truth association. We will define scoring criteria for track purity in dense target environments. In principle, track purity can be used to determine

---

2 O.E. Drummond, *Multiple-Object Estimation*, UCLA Ph.D. Dissertation, 1975. Xerox University Microfilms No. 75-26, 954.

3 O.E. Drummond, *Multiple Target Tracking Lecture Notes*, UCLA, Oct. 1985; Revised 10 December 1990, Technology Training Corporation, Torrance, CA.

4 S.S. Blackman, *Multiple Target Tracking with Radar Applications*, Artech House, Dedham, MA (1986).

5 O.E. Drummond, "The Algorithm Development Challenge of Tracking the SDI Dense Threat", *IST Workshop on SDI: BM/C³*, IDA. Alexandria, VA, 24 November 1987.

track-to-truth associations but in dense target environments and for some MTT algorithms the concept of track purity loses some of its meaning.

We suggest a method for track-to-truth association based on a global nearest neighbor assignment approach. At each of the designated evaluation times, a global nearest neighbor assignment algorithm is executed to *uniquely* associate tracks and targets. After tracks and truth have been associated, we can evaluate performance criteria for the two functions of a multiple target tracking algorithm:

1. *Data association*. This function selects the observations to be used by the track filter to update the state estimate. Its measures of effectiveness will be track purity and misassociation. They measure the consistency with which a track is updated with measurements from a single target or a set of targets, respectively; and,

2. *Track filter*. This function transforms sensor measurements into estimates of the target's state, usually the target's trajectory described by position, velocity, acceleration, etc, and the target's state estimation error covariance. The distance between the state estimate and the true state and the credibility of the filter calculated covariance matrix measure the performance of the tracking filter, which is affected by data misassociation and other errors.

7.1 AN ASSIGNMENT APPROACH TO TRACK-TO-TRUTH ASSOCIATION

A method for associating track-to-truth by assignment algorithm was introduced[1, 2] in connection with the so-called "basis free" estimation, in which none of the information sets used in the estimation process is employed to distinguish one target estimate from another. This approach has been adapted by the SDI Panels on Tracking. In the assignment approach to performance evaluation, state estimates are paired to their nearest true target state using a global nearest neighbor criterion, which is equivalent to finding the most probable global hypothesis.

The implementation of the approach is conceptually simple. The estimates are treated as one data set and the truth as another. At each evaluation time an assignment algorithm is applied to these two data sets so that there is a *unique* assignment of tracks to truth. As a consequence, no track (state estimate) is assigned to more than one true target and no true target is assigned to more than one track.

Local nearest neighbor assignments of track-to-truth are to be avoided. Target-based local nearest neighbor, in which each target is independently associated with the nearest track does not penalize missed tracks: each target can always find a track because tracks can be assigned to more than one target. Similarly, track-based local nearest neighbor, in which each track is independently associated with the nearest target, does not penalize redundant or spurious tracks: each track can always find a target because targets can be assigned to more than one track. Because some tracks or targets are not assigned in the global nearest neighbor assignment algorithm, missed, redundant, and spurious tracks can be identified.

In order to perform the assignment, a cost (goodness-of-fit) must be computed for each candidate assignment: a pair of track and true target state. The optimal global assignment finds the combination of pairs that provides the minimum sum of costs. Each cost can be viewed as minus the log of the likelihood that the estimated target is the true target for a given pairing. In most cases this can be computed as a chi-square statistic. The parameter or state vector used in the cost can contain only position information or additional elements (such as velocity estimates and intensity) depending on the application.

Simplifications to the chi-square statistic can reduce the computing cost. For example, the off diagonal terms in the covariance matrix can be omitted so that the chi-square statistic is simply a weighted sum of squares. Furthermore, this approach to performance evaluation employs methods akin to the "assignment approach" (global nearest neighbor) for multiple target tracking (MTT). Many of the specific techniques used in MTT, such as gating, "binning," and "sparse" assignment algorithms, can be used to reduce the cost of performance evaluation when using the described assignment approach. However, when using the sparse version of the assignment algorithm, such as the JVC[6] or the Stephens-Krupa sparse Munkres,[7] it is important to pre-condition the input to the assignment algorithm to ensure that there is a feasible solution.[2, 9]

7.2 DATA ASSOCIATION AND TRACK PURITY

With the assigned track-truth pairs, data association, track purity, and track accuracy can be evaluated. Kovacich and Chong have described a scoring method for data

---

[6] O.E. Drummond, D.A. Castanon, and M.S. Bellovin, "Comparison of 2-D Assignment Algorithms for Sparse, Rectangular, Floating Point, Cost Matrices," *Journal of the SDI Panels on Tracking*, Issue No. 4/1990, 15 Dec. 1990, pp. 4-81 to 4-97.

[7] P.A. Stephens and N.R. Krupa, *A Sparse Matrix Technique for the Munkres Algorithm*, 1979 Summer Computer Simulation Conference, Toronto, Canada, July 1979, pp. 44-47.

association and track purity that develops its own assignment of track-to-truth.[8] They define *loose sense track purity* over a specified time interval as a measure of effectiveness for data association to evaluate the degree of consistent updating with measurements from the same set of targets. This would be more appropriate in environments where often a group of closely spaced objects generate a single observation. They define *strict sense track purity* over a specified time interval to evaluate the degree to which a track uniquely represents a single target. This would be more appropriate to evaluating tracking in support of target discrimination.

Consider the three different classes of data association algorithms discussed in Section B: unique assignments to global or local nearest neighbors; probability data association and joint probability data association (PDA/JPDA); and multiple hypothesis tracking (MHT). In nearest neighbor assignment, a track consists of a sequence of individual observations, one from each frame of data, each arising from none, one, or more targets. In the PDA/JPDA approach, at each update time, the set of observations with feasible association to the track are associated with the track in proportion to their probability of association. A composite state estimate results which is a weighted sum of the observations. A PDA/JPDA track, therefore, consists of a sequence of sets of observations, a set for each update time, each set consisting of individual observations, each observation arising from none, one, or more targets.

Now consider Reid's MHT[9] approach which generates at each update time a set of competing data association hypotheses that Reid called cluster hypotheses. Each cluster hypothesis consists of one possible set of observation-to-track associations; a particular observation-to-track association may appear in many different cluster hypotheses. Therefore, Reid's MHT produces tracks that consist of a sequence of individual observations, one from each frame of data, each arising from none, one, or more targets, similar to nearest neighbor assignment. Hypotheses or tracks considered unlikely, say those below some threshold, are dropped, while those that are "similar" according to some criteria are combined.

Kovacich and Chong develop different track purity scores for what they term as single frame assignment logic, PDA/JPDA assignment logic, and MHT approaches. For

---

8 Michael Kovacich and Chee Chong, "Definition of Track Purity," *The Proceedings of the SDI Tracking Panels*, Issue No. 3/1989, pp. 1-13 to 1-19, 1 July 1989.

9 Donald B. Reid, "An Algorithm for Tracking Multiple Targets," *IEEE Transactions on Automatic Control*, Vol. AC-24, No. 6, December 1979, pp. 843-854.

simplicity we will focus on track purity measures for what they refer to as single frame assignment logic and apply it to the Example of Fig. 7-1 without regard to the data association algorithm actually used. Kovacich and Chong's other methods are described in the appendix to this Section.

TRACK	OBSERVATION SET				
1	(ab)	(ab)	a	b	ø
2	ø	ø	b	a	(ab)
3	(cd)	(cd)	(cd)	(cd)	ø
4	e	(ef)	e	f	FA
5	f	ø	f	e	f
6	ø	ø	ø	ø	e

Figure 7-1. Example of a Set of Hypothetical Track Histories. The letters indicate the target identifications that contributed to the observations used by the tracks. Parentheses indicate target composition of unresolved closely spaced objects.

Kovacich and Chong divide track purity for single frame assignment logic into three steps. For this purpose, Kovacich and Chong denote the sequence of individual observations in a track as the Measurement Set for that track. The set of targets that generate an observation they denote as the Target Set for the observation. In the first step, compute a score function that serves as the criteria by which tracks are associated to truth:

Step 1 Given track i and target j, for strict sense track purity compute

$$SSS(i,j) = NM(i,j)/NMEAS(i)$$

where

NM(i,j) = number of observations in the Measurement Set for track i whose Target Set contains target j.

NMEAS(i) = number of observations in the Measurement Set for track i.

For loose sense track purity, given track i and target j, compute

$$LSS(i,j) = NM(i,j)/NTGTS(i)$$

where

$$NTGTS(i) = \sum_k NTGTS(i,k)$$

NTGTS(i,k) = number of targets in the target set for the observation on the kth observation opportunity.

The results of applying these to the Example are in Fig. 7-2. It seems that neither properly accounts for a failure to associate any observation in the frame with the track, for example, because of an empty gate. We suggest an alternative score, given track i and target j

$$SA(i,j) = NM(i,j)/NMEASF(j)$$

where

NMEASF(j) = number of frames that target j was detected in the field of view during the time interval over which track purity is being evaluated.

This score function is also in Fig. 7-2.

In the second step, Kovacich and Chong associate targets to tracks in strict sense track purity using the SSS(i,j) as the score matrix in an assignment algorithm. Observe that many assignment algorithms are designed to minimize the total cost (an undesirable quantity) rather than maximize the total profit/benefit (a desirable quantity). Distance is a cost but track purity is a profit/benefit. Thus, when assigning tracks to truth based on track purity, which is to be maximized, the individual profits have to be converted to cost, for instance, by changing the sign. At the conclusion of this step, each track i will be associated with a unique target, A(i), or none.

In loose sense track purity, Kovacich and Chong prescribe the use of LSS(i,j), as well as other criteria--which they do not specify--to determine the set of targets to associate with the track. They define ASET(i) as the set of targets associated with track i in this manner.

We input the score functions, SSS(i,j), LSS(i,j), and SA(i,j) to both JVC and Stephens and Krupa sparse Munkres assignment algorithms, the results for which are in Fig. 7-3. There are other "optimal" solutions besides those in Fig. 7-3: Note the ambiguous track-to-truth association among tracks 1 and 2 and targets a and b. Note that we used a global nearest neighbor assignment algorithm with LSS because the "other criteria" were not specified.

$$SSS(i,j) = \begin{bmatrix} 3/4 & 3/4 & 0 & 0 & 0 & 0 \\ 2/3 & 2/3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4/4 & 4/4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3/5 & 2/5 \\ 0 & 0 & 0 & 0 & 1/4 & 3/4 \\ 0 & 0 & 0 & 0 & 1/1 & 0 \end{bmatrix}$$

$$LSS(i,j) = \begin{bmatrix} 3/6 & 3/6 & 0 & 0 & 0 & 0 \\ 2/4 & 2/4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4/8 & 4/8 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3/5 & 2/5 \\ 0 & 0 & 0 & 0 & 1/4 & 3/4 \\ 0 & 0 & 0 & 0 & 1/1 & 0 \end{bmatrix}$$

$$SA(i,j) = \begin{bmatrix} 3/5 & 3/5 & 0 & 0 & 0 & 0 \\ 2/5 & 2/5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4/4 & 4/4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3/5 & 2/5 \\ 0 & 0 & 0 & 0 & 1/5 & 3/5 \\ 0 & 0 & 0 & 0 & 1/5 & 0 \end{bmatrix}$$

Figure 7-2. Track Purity Input Matrices for Assignment Algorithms.

STRICT SENSE TARGET-TRACK ASSIGNMENTS

1	↔	a
2	↔	b
3	↔	c
4	↔	-
5	↔	f
6	↔	e

LOOSE SENSE TARGET-TRACK ASSIGNMENTS

1	↔	a
2	↔	b
3	↔	c
4	↔	-
5	↔	f
6	↔	e

TARGET-TRACK ASSIGNMENTS BASED ON SA(i,j)

1	↔	a
2	↔	b
3	↔	c
4	↔	e
5	↔	f
6	↔	-

Figure 7-3. Optimal Assignments of Tracks to Truth Using Both the JVC and the Stephens and Krupa Sparse Munkres Assignment Algorithms, Which Arrived at the Same Answer. Note that there are other "optimal" solutions, that is, alternate solutions with the same value for the optimization criteria.

Last, in the third step, compute the track purity:

<u>Step 3</u> For track i compute the strict sense track purity

$$TPSS(i) = NPURE(i,A(i))/NMEAS(i)$$

where

TPSS(i) = the strict sense track purity for track i.

NPURE(i,A(i)) = number of *pure* observations in the measurement set for track i generated by target A(i), where a pure observation is one whose source is a single target;

for track i compute the loose sense track purity

$$TPLS(i) = \sum_{j \in ASET(i)} LSS(i,j).$$

For comparison we used SA(i,j) for both the cost in the assignment algorithm and to computer the track purity. The various track purity scores are in Fig. 7-4.

TRACK #	STRICT SENSE PURITY TPSS	LOOSE SENSE PURITY TPLS	LOOSE SENSE PURITY SA
1	.25	.50	.60
[illegible]	.33	.50	.40
[illegible]	0	.50	1.0
4	0	0	.60
[illegible]	.75	.75	.60
[illegible]	1.0	1.0	0

Figure 7-4. Track Purity Scores

We feel that for performance evaluation at each evaluation time the association of tracks to truth should be done once and for all using a global nearest neighbor assignment algorithm based on a chi-square distance measure. These pairings would then be used to evaluate track purity, misassociations, state estimation errors, and the credibility of the filter calculated covariance. With this approach, the assignment step in Kovacich and Chong, Step 2 would be obviated.

Evaluation of the data association function in support of target discrimination should not penalize for unassociated observations but should penalize for associations to unresolved closely spaced objects, the wrong object, or false alarms. If one were to make assignments as per Kovacich and Chong for strict sense track purity, then the method should be revised to use TPSS(i,j) rather than SSS(i,j).

Evaluation of misassociation performance should penalize for unassociated observations as well as associations to the wrong object and false alarms and should not penalize for associations to unresolved closely spaced objects. If one were to make assignments as per Kovacich and Chong for loose sense track purity, then the method should be revised to use SA(i,j) rather than LSS(i,j).

7.3 STATE ESTIMATION ACCURACY AND FILTER COVARIANCE CREDIBILITY

One straightforward measure of state estimation accuracy is the error magnitude. Direct measures include position error, velocity error, etc., defined in the usual manner as the magnitude of the position component, velocity component, etc., of the state estimation error vector. Clearly, the smaller the errors the better but, for tracking considered independently of other functions, how small is small enough? Because of the complicated statistics describing position error, velocity error, etc. (square root of a sum of squared jointly normally distributed random variables), we cannot easily determine small enough in a statistical sense.

State estimation error magnitude is related to bias error for which we will define statistical tests. We will also define statistical tests to measure filter covariance credibility, which occurs when the filter calculated covariance substantially differs from the actual covariance in the state estimation error. Together bias and filter covariance credibility measure the accuracy and consistency of the modeling assumptions of the tracking filter in relation to the actual target dynamics and the effects of measurement errors, misassociations, and unresolved closely spaced objects. We will define tests for statistical significance for each and determine confidence intervals to specify when the state estimates are good and the filter calculated covariance is credible.

To test statistical significance in single target tracking we would collect sample statistics on the stochastic processes being modeled by performing many Monte Carlo runs of the identical scenario for a single target and its single track. The presence of misassociations in multiple target tracking can introduce ambiguities in isolating the same single track-target pair over many Monte Carlo runs. In a dense environment, the source of the measurements in a track could vary with each run, as may the association of the track to truth. For multiple target tracking simulations, one could perform one run of a large-scale Monte Carlo simulation in place of repeating the target scenario over many Monte Carlo runs. The collection of the M independent track-target pairs in the single Monte Carlo run of a dense target tracking problem forms an ensemble of sorts over which we can compute sample statistics for the stochastic processes being modeled. A weakness of this approach pertains to differences in the covariance matrices across the M tracks: all the tracks will not be of the same age and model nonlinearities and variations in local observation densities can cause differences in track covariances.

7.3.1 State Estimation Error

Given the assignment of tracks to truth for various evaluation times, the accuracy of the track state estimate can be evaluated. The type of estimates (predicted, filtered, or smoothed) must be selected as well as the evaluation times. For each of the M track-target pairs that are associated, compute the state estimation error as the difference between the true target state x(k) and the track estimated state at time k given measurements to time n:

$$\tilde{x}(k|n) = x(k) - \hat{x}(k|n) \qquad (7\text{-}1)$$

The usual magnitude of the position estimation error, also known as the root-sum-square (RSS) error, is computed by taking the square root of the sum of the squared position components of the state estimation error vector. The RSS velocity error (and acceleration, etc.) is similarly computed. The sample cumulative probability distribution of the error magnitudes can be plotted for all the target-track pairs. The missed, redundant, and spurious tracks cannot be readily included in this plot unless default values are provided for the tracks and the targets that are not paired, that is, left unassigned by the global nearest neighbor assignment algorithm.

Both the mean and median of the RSS errors can be computed. Before computing these parameters, especially the mean, it may be acceptable to edit out the M worst tracks. If this editing is allowed for the system being evaluated, then the value of M should be specified in advance and the worst tracks should include all the spurious and missed tracks. The requirement might also establish that a specified percentage of the tracks have an error less than a given magnitude.

7.3.2 State Estimation Bias

To test the bias of the state estimate, which is a vector, test each component of the state estimation error individually. Under the hypothesis that the state estimate is unbiased, and assuming that the error is normally distributed and, therefore, each component, indexed by the subscript i, is also normally distributed:

$$e_i = \tilde{x}_i(k|n) \sim N[0, P_{ii}(k|n)] \quad . \tag{7-2}$$

Assuming that each of the M state estimation errors is independent with identical mean and covariance, we can compute the sample mean and sample variance to form a test statistic as follows.

The sample mean of the ith component of the state estimation error over the M track-target pairs is defined as

$$\bar{e}_i = \frac{1}{M}\sum_{j=1}^{M}[e_i]_j \tag{7-3}$$

where j indexes the M track-target pairs and the time indices have been dropped for clarity. The sample variance is defined as

$$s_{ii} = \frac{1}{M-1}\sum_{j=1}^{M}([e_i]_j - \bar{e}_i)^2 \quad . \tag{7-4}$$

We can form a chi-squared distributed random variable with M-1 degrees of freedom as

$$\frac{s_{ii}(M-1)}{P_{ii}} \sim \chi^2_{M-1} \quad . \tag{7-5}$$

Now define a zero mean, unit variance normally distributed random variable

$$z = e_i/\sqrt{P_{ii}} \sim N(0,1) \tag{7-6}$$

so that a t distributed random variable can be defined

$$
\begin{aligned}
t_{M-1} &= \frac{z}{\sqrt{\chi^2_M / (M-1)}} = \frac{e_i}{\sqrt{P_{ii}}} \cdot \frac{\sqrt{P_{ii}}}{\sqrt{s_{ii}}} \\
&= e_i / \sqrt{s_{ii}} \ .
\end{aligned}
\tag{7-7}
$$

The ith component of the state estimation error is unbiased when

$$
\text{Prob}\left[\,\left| e_i / \sqrt{s_{ii}} \right| \le t_{\alpha/2}\,\right] = 1 - \alpha \quad \text{e.g., } \alpha = .05
\tag{7-8}
$$

where for a given confidence level, $1 - \alpha$, $t_{\alpha/2}$ follows from the properties of the t-distribution function.

7.3.3 Filter Covariance Credibility

If the estimate is unbiased then we can test for filter covariance credibility by forming another chi-squared random variable following Bar-Shalom and Fortmann.[10] For each of the M track-target pairs, compute the state estimation error chi-squares for the entire state, position only, or velocity only, etc. For instance, the full state chi-square is:

$$
\begin{aligned}
\varepsilon(k \mid n) &= \tilde{x}^T(k \mid n)\, P^{-1}(k \mid n)\, \tilde{x}(k \mid n) \\
&= \chi^2_{n_x}
\end{aligned}
\tag{7-9}
$$

where P is the filter calculated covariance matrix and n_x is the chi-square's number of degrees of freedom, i.e., the dimension of the full state. We test the credibility of the filter-calculated covariance P by forming an ensemble average chi-square as the sample mean over the M individual track-target pair chi-squares, which are assumed independent.

$$
\bar{\varepsilon}(k \mid n) = \frac{1}{M} \sum_{j=1}^{M} \varepsilon^j(k \mid n)
\tag{7-10}
$$

where j indexes the M track-target pairs. Then

---

[10] Yaakov Bar-Shalom and Thomas E. Fortmann, *Tracking and Data Association*, Academic Press, Orlando, Florida (1988).

$$M\bar{\epsilon}(k \mid n) \sim \chi^2_{M n_x} \qquad (7\text{-}11)$$

and the confidence interval is computed from

$$\text{Prob}\{M\bar{\epsilon}(k \mid n) \in [r_1, r_2]\} = 1-\alpha \quad ,\text{e.g. } \alpha = .05 \qquad (7\text{-}12)$$

For a given confidence level, $1-\alpha$, r_1 and r_2 will follow from the chi-square distribution with M_{n_x} degrees of freedom. The state estimation errors are consistent with the filter calculated covariances if

$$\bar{\epsilon}(k \mid n) \in [r_1/M, r_2/M] \quad . \qquad (7\text{-}13)$$

In addition to the scores for performance evaluation described, additional measures of performance will be needed. Some additional tracking scores of interest include the number of missing tracks, redundant tracks, and spurious tracks. These can be readily computed after the tracks are uniquely assigned to the true targets. Performance evaluation typically includes scores of track length and length of time to initiate a track (track acquisition). The various tracking stages of initiation, maintenance, and termination for a single sensor or multiple sensors as appropriate must also be considered.

A number of tracking scores have been listed by Willman[11] and others.[12] There has been little in the literature, however, on how to assign tracks to target truth in order to fairly evaluate diverse tracking algorithms and adequately penalize missing, redundant, and spurious tracks. This is not a significant issue for tracking a single target or in a sparse environment. As system requirements call for tracking multiple targets in a moderate to dense environment or with low observables, this becomes a major concern in performance evaluation. The evaluation methods described are designed to address these challenging conditions by uniquely assigning tracks to true targets.

---

11 W.W. Willman, *Some Performance Results for Recursive Multitarget Correlator-Tracker Algorithms*, Naval Research Laboratory, Washington, D.C., NRL Report 8423, July 1980.

12 Op. cit., Bar-Shalom and Fortmann and Blackman (see p. 7-14).

7.4 CONCLUSIONS

We have presented scoring methods being pursued by the SDI Panels on Tracking. These methods attempt to represent the needs of a broad variety of MTT algorithms to be judged fairly. The scoring methods are part of an ongoing process and are not to be considered the last word on the subject. The Panels have suggested that for the time being these methods be adopted throughout the SDI tracking community, particularly for the Surveillance Testbed activity.

APPENDIX, CHAPTER 7

The following is from Kovacich and Chong.[13]

- **Strict Sense Track Purity for JPDA**

Step 1 Given track i and target j, compute

S(i,j)	=	P(i,j) / NMEAS(i)

where

P(i,j)	=	$\sum_k P(i,j,k)$
P(i,j,k)	=	Probability that the composite measurement for track i on measurement opportunity k contains target j.
NMEAS(i)	=	Number of composite measurements in the measurement set for track i.

Step 2 Uniquely assign targets using the S(i,j) matrix as the score or cost matrix. At the conclusion of this step, each track i will be assigned to a unique target A(i).

Step 3 Assign a measurement to the track for each measurement opportunity in which a measurement occurred. For example, assign the measurement that has the highest weight in the composite measurement. Denote the measurement assigned to the track the Assigned Measurement.

Step 4 Compute track purity for track i:

TPSS(i)	=	NPURE(i,A(i))/NMEAS(i)

where

NPURE(i,A(i))	=	Number of *pure* Assigned Measurements in the Measurement Set for track i generated by A(i).

13 Op. cit., Bar-Shalom and Fortmann and Blackman (see p. 7-14).

NMEAS(i)	=	Number of measurements in the measurement set for track i.

- **Strict Sense Track Purity for Multiple Hypothesis Logics with Composite Tracks**

Step 1 Select one of the feasible tracks to represent the composite track. For example, the feasible track that has the highest weighting could be selected.

Step 2 Perform Steps 1, 2, and 3 in the Track Purity calculation for Single Frame Assignment Logics using the selected feasible track. The track purity for composite track i is defined to be the track purity of the selected feasible track.

- **Loose Sense Track Purity for the JPDA Logic**

Step 1 Given track i and target j, compute

$$S(i,j) = P(i,j) / NTGTS(i)$$

where

$$P(i,j) = \sum_k P(i,j,k)$$

P(i,j,k)	=	Probability that the composite measurement for track i on measurement opportunity k contains target j.

$$NTGTS(i) = \sum_k NTGTS(i,k)$$

NTGTS(i,k)	=	Number of targets in the target set for the measurement on the kth measurement opportunity.

Step 2 Using S(i,j), as well as other criteria, determine the set of targets to assign to the track. Let ASET(i) specify the set of targets assigned to track i.

Step 3 Compute track purity for track i

$$TPLS(i) = \sum_{j \in ASET(i)} S(i,j)$$

- **Loose Sense Track Purity for Multiple Hypothesis Logics with Composite Tracks**

Step 1 Perform Steps 1, 2, and 3 in the Track Purity calculation for Single Frame Assignment Logics to calculate a track purity for each feasible track:

$TP(i,j)$ = Track purity of feasible track in composite track i.

Step 2 Compute the track purity for the composite track i as follows:

$$TP(i) = \sum_j P(i,j) * TP(i,j)$$

where

$P(i,j)$ = Probability that the composite track i contains feasible track j.